

Luiz Miranda Cavalcante Neto

**Recuperação de trajetória de ponta de caneta
em assinaturas *offline* com referencial *online***

Brasil

Janeiro, 2017

Luiz Miranda Cavalcante Neto

**Recuperação de trajetória de ponta de caneta em
assinaturas *offline* com referencial *online***

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica (PROEE) da Universidade Federal de Sergipe, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia Elétrica.

Universidade Federal de Sergipe

Departamento de Engenharia Elétrica

Programa de Pós-graduação em Engenharia Elétrica

Orientador: Jugurta Rosa Montalvão Filho

Brasil

Janeiro, 2017



UNIVERSIDADE FEDERAL DE SERGIPE
PRÓ-REITORIA DE PÓS-GRADUAÇÃO E PESQUISA
COORDENAÇÃO DE PÓS-GRADUAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA-PROEE

TERMO DE APROVAÇÃO

**“Recuperação de Trajetória de Ponta de Caneta em
Assinaturas Offline com Referencial Online”**

Discente:

Luiz Miranda Cavalcante Neto

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Sergipe, como parte dos requisitos exigidos para a obtenção do título de Mestre em Engenharia Elétrica.

Aprovada pela banca examinadora composta por:

Prof. Dr. Elyson Ádan Nunes Carvalho (PROEE/UFS)
Presidente

Prof. Dr. Jânio Coutinho Canuto (PROEE/UFS)
Examinador Interno

Prof. Dr. Leonardo Nogueira Matos (PROCC/UFS)
Examinador Externo

Luiz Miranda Cavalcante Neto
Candidato

Cidade Universitária “Prof. José Aloísio de Campos”, 30 de janeiro de 2017.

**FICHA CATALOGRÁFICA ELABORADA PELA BIBLIOTECA CENTRAL
UNIVERSIDADE FEDERAL DE SERGIPE**

C376r Cavalcante Neto, Luiz Miranda
Recuperação de trajetória de ponta de caneta em assinaturas
offline com referencial *online* / Luiz Miranda Cavalcante Neto ;
orientador Jugurta Rosa Montalvão Filho. – São Cristóvão, 2017.
53 f. ; il.

Dissertação (mestrado em Engenharia elétrica) – Universidade
Federal de Sergipe, 2017.

1. Engenharia elétrica. 2. Assinaturas. 3. Assinaturas digitais.
4. Identificação biométrica. I. Montalvão Filho, Jugurta Rosa,
orient. II. Título.

CDU: 621.3:004.932.2

Agradecimentos

Agradeço à CAPES por financiar dois anos do desenvolvimento deste trabalho. Ao meu orientador, Jugurta, por sempre me ajudar a resolver os problemas que eu levava a ele. Aos meus pais, Luiz e Flávia, por serem meus pais. À minha namorada, Ana, por me suportar por tanto tempo. Aos meus amigos que sempre que nos encontramos convertem um dia ruim em bom. E aos meus bichinhos que tornam minha casa um lugar melhor de viver.

Resumo

A assinatura manuscrita é uma forma de identificação pessoal amplamente aceita, tanto social como juridicamente, utilizada há séculos para autenticar documentos como cheques bancários, cartas, contratos e todo tipo de serviço que necessite prova de autoria. Ao assinar, um indivíduo insere uma grande quantidade de informação para transformar o movimento de sua mão em um elemento identificador. Velocidade de escrita, trajetória percorrida, inclinação da caneta, pressão aplicada, todos esses dados são articulados (na forma de variáveis latentes) para resultar em uma figura estática no documento assinado. Essa dissertação investiga o problema de extração de trajetória da caneta que gera essa assinatura a partir da imagem estática resultante. Para isso, o trabalho foi dividido em três passos principais que são: (i) computar o esqueleto da assinatura *offline*; (ii) extrair suas características utilizando um conceito abordado neste trabalho chamado UCSS, e, com o auxílio de assinaturas *online* registradas previamente; (iii) estimar o caminho que a caneta percorreu sobre o esqueleto. Em cada um desses passos, foi feita uma revisão de trabalhos relevantes sobre os temas para só então iniciar as implementações. Foram realizados três experimentos durante este trabalho, o primeiro foi feito com o objetivo de comparar os resultados obtidos com o algoritmo desenvolvido e os resultados obtidos em um trabalho de referência, os outros dois foram realizados durante a produção de um artigo destinado a publicação anexado a este trabalho.

Palavras-chave: Recuperação de Trajetória. Assinatura *online*. Assinatura *offline*. Esqueletização. Extração de Características. Verificação de Assinatura. Biometria.

Abstract

The handwritten signature is a form of personal identification widely accepted, both socially and legally, used for centuries to authenticate documents such as bank checks, letters, contracts and any type of service that requires proof of authorship. When signing, an individual inserts a large amount of information to transform the movement of his hand into an identifying element. Writing speed, trajectory traversed, pen inclination, applied pressure, all these data are articulated (in the form of latent variables) to result in a static figure in the signed document. This dissertation investigates the problem of trajectory extraction of the pen that generates this signature from the resulting static image. For this, the work was divided in three main steps that are: (i) compute the skeleton of the offline signature; (ii) extract its characteristics using a concept addressed in this work called UCSS, and, with the help of previously recorded online signatures, (iii) estimate the path that the pen traveled over the skeleton. In each of these steps, a review of relevant work on the themes was done prior to begin implementation. Three experiments were carried out during this work, the first one was done with the objective of comparing the results obtained with the developed algorithm and the results obtained in a reference work, the other two were realized during the production of an article destined to the publication attached to this job.

Keywords: Trajectory Recovery. Online Signatures. Offline Signatures. Skeletonization. Feature Extraction. Signature Verification. Biometry.

Lista de ilustrações

Figura 1	– (a) Exemplo de assinatura com formato complexo. (b) Exemplar <i>online</i> da mesma assinatura em (a). Adaptado de Nel (2005).	3
Figura 2	– Esquema de um sistema de recuperação de trajetória padrão. Adaptado de Nguyen e Blumenstein (2010).	6
Figura 3	– Organização dos <i>pixels</i> para o algoritmo de esqueletização (GONZALEZ; WOODS, 2007).	8
Figura 4	– a) Assinatura original em preto e branco (formato de “fita”). b) Resultado do processo de esqueletização usando o algoritmo clássico. Os círculos vermelhos apontam alguns artefatos resultantes deste processo (formato de “linha”).	9
Figura 5	– a) Esqueleto primário sobre o texto original obtido via MAT. b) Esqueleto com PMAT aplicada. c) Aplicação da CPMAT sobre o esqueleto. Adaptado de (BOCCIGNONE et al., 1993)	10
Figura 6	– a) Esqueleto de inicialização feito usando o método clássico. b) Esqueleto inicial após a etapa de Ajuste e Suavização. c) Aplicação da etapa de Reconstrução. d) Ajustes e Suavizações finais. Adaptado de (KÉGL; KRZYZAK, 2002)	11
Figura 7	– Método de esqueletização em Dawoud e Kamel (2003), as linhas representam as iterações do algoritmo e as colunas os passos em cada iteração. a) <i>Pixels</i> em preto são os selecionados na iteração (limiarização) anterior, <i>pixels</i> em cinza são os selecionados na iteração atual. b) Tratamento dos <i>pixels</i> do esqueleto resultante da iteração anterior com os <i>pixels</i> selecionados na iteração atual. c) Esqueleto resultante na iteração atual.	12
Figura 8	– (a) Assinatura binarizada. (b) Extração das bordas da assinatura. (c) Resultado da Triangulação Limitada de Delaunay.	13
Figura 9	– a) Ampliação de um pedaço da assinatura segmentada e identificação dos tipos de triângulos formados. Em vermelho os pontos característicos de cada um dos triângulos. Os “x” representam os pontos médios das arestas internas do TN que geram seu ponto característico. b) Esqueleto primário da assinatura.	14
Figura 10	– (a) Polígono testado em preto e seu círculo característico em roxo. (b) União dos dois polígonos por não suprir a condição de estabilidade	15

Figura 11 – (a) Em azul os pontos característicos dos polígonos vizinhos do polígono de junção testado (em vermelho) e em preto o ponto P que minimiza a distância entre as três retas. Só foram usados dois pontos para encontrar os parâmetro da reta r_3 pois o terceiro vizinho era um polígono de junção.(b) União dos polígonos na direção do ponto P	16
Figura 12 – (a) Resultado do processamento dos polígonos (b) Esqueleto Final	16
Figura 13 – Imagem adaptada da identificação das regiões ambíguas e pontos característicos no trabalho de Huang, Rong e Bian (1995).	18
Figura 14 – (a) Esqueleto de uma palavra escrita a mão com os nós marcados de 1 a 7. (b) Ampliação do nó 1. Adaptado de (QIAO; NISHIARA; YASUHARA, 2006)	19
Figura 15 – Ilustração da modelagem de um esqueleto em um HMM. a) O esqueleto original representado de maneira contínua, por critério estético. b) Os <i>pixels</i> do esqueleto organizados em um grafo ligando <i>pixels</i> vizinhos. c) Esqueleto modelado como um HMM, as setas representam as possíveis transições que podem ocorrer. Adaptado de (NEL; PREEZ; HERBST, 2005).	19
Figura 16 – Classificação das regiões de curvatura máxima segundo Plamondon e Privitera (1999).	20
Figura 17 – Identificação das regiões ambíguas. a) Região de cruzamento. c) Região de bifurcação. c) Pontos de máxima curvatura nas regiões ambíguas gerando uma área de decisão. Adaptado de (BAATI et al., 2005).	21
Figura 18 – Esqueleto da assinatura presente na Figura 1 com os UCSS extraídos representados pelas diferentes cores e uma aproximação ilustrando o registro dos segmentos nos dois sentidos.	22
Figura 19 – Ilustração das medidas usadas para calcular a função de continuidade. Marcados com os “x” estão os pontos que ancoram os segmentos usados para essa ilustração, a região de encontro é separada à direita e suas medidas são explicitadas. Adaptado de (BOCCIGNONE et al., 1993).	25
Figura 20 – Ilustração da medida de Distância Acumulada (DA) para: a) Um segmento UCSS e b) Um segmento de uma assinatura <i>online</i>	27
Figura 21 – a) Assinatura <i>offline</i> para ter trajetória recuperada. b) Seu referencial dinâmico. c) Um dos Exemplos Dinâmicos disponíveis na base de dados.	27
Figura 22 – a) Em laranja o Exemplo Dinâmico que será testado e em azul o esqueleto da assinatura <i>offline</i> . b) Ajuste de translação e c) Ajuste de rotação e escala.	28
Figura 23 – Esquemático do funcionamento do sistema completo.	30
Figura 24 – Ilustração do protocolo de avaliação.	33

Figura 25 – a) Assinatura <i>offline</i> com falha no traço marcado pelo círculo. b) Referencial Dinâmico de (a).c) Trajetória recuperada erroneamente devido ao corte no traço.	34
Figura 26 – Exemplo de um dos cartões distribuídos para os voluntários utilizados para obter a MOS	36
Figura 27 – Exemplo de um dos cartões distribuídos para os voluntários utilizados para obter a MOS	37

Lista de tabelas

Tabela 1 – Comparação entre os resultados de similaridade média do trabalho de Nel, Preez e Herbst (2005) com o sistema proposto nesta dissertação usando: o método de esqueletização descrito na seção 2.3 e o método de esqueletização clássica.	32
Tabela 2 – Resultado de cada método após 50 execuções independentes. Cada execução corresponde a um grupo aleatório de 5 assinaturas genuínas selecionadas para o modelo de cada usuário.	37

Lista de abreviaturas e siglas

CRF	Conditional Random Field
DA	Distância Acumulada
DTW	Dynamic Time Warping
ECR	Edge Continuity Relation
EER	Equal Error Rate
HMM	Hidden Markov Model
MAT	Medial Axis Transform
TI	Triângulo Isolado
TJ	Triângulo de Junção
TN	Triângulo Normal
TT	Triângulo de Terminação
UCSS	Unambiguously Connected Skeleton Segments

Sumário

1	INTRODUÇÃO	1
1.1	Extração de trajetória e suas dificuldades	2
1.2	Objetivos	5
1.3	Organização	6
2	ESQUELETIZAÇÃO	7
2.1	Esqueletização Clássica	7
2.2	Pseudo-Esqueletização	10
2.3	Algoritmo Implementado	12
2.3.1	Processamento de Polígonos de Terminação	14
2.3.2	Processamento de Polígonos de Junção	14
3	EXTRAÇÃO DE CARACTERÍSTICAS	17
3.1	Trabalhos relevantes	17
3.2	Método Proposto	20
3.2.1	Definição de segmentos conectados sem ambiguidades - UCSS	21
3.2.2	Aplicação dos UCSS para verificação de assinaturas <i>offline</i>	22
4	RECONSTRUÇÃO DA TRAJETÓRIA DA ASSINATURA	24
4.1	Trabalhos relevantes	24
4.2	Método Proposto	26
4.2.1	Passo de inicialização	28
4.2.2	Passo de reconstrução	29
5	RESULTADOS E DISCUSSÕES	31
5.1	Experimentos de Reconstrução de Trajetória	31
5.1.1	Protocolo de Avaliação da Reconstrução	32
5.1.2	Resultados do experimento de reconstrução	32
5.2	Experimentos de Verificação Biométrica	35
5.2.1	Resultados do Experimento de Verificação Biométrica	37
6	CONCLUSÕES	39
	REFERÊNCIAS	41

ANEXOS	45
ANEXO A – ARTIGO PRODUZIDO PELO AUTOR	46

1 Introdução

A assinatura manuscrita é uma forma de identificação pessoal amplamente aceita, tanto social como juridicamente, utilizada há séculos para autenticar documentos como cheques bancários, cartas, contratos e todo tipo de serviço que necessite prova de autoria. Ao assinar, um indivíduo insere uma grande quantidade de informação para transformar o movimento de sua mão em um elemento identificador. Velocidade de escrita, trajetória percorrida, inclinação da caneta, pressão aplicada, todos esses dados são articulados (na forma de variáveis latentes) para resultar em uma figura estática no documento assinado (NEL, 2005). Essa dissertação investiga o problema de extração de trajetória da caneta que gera essa assinatura a partir da imagem estática resultante. Para isso, podemos assumir que existem duas formas gerais de registrar e analisar assinaturas, o modo *offline* e o modo *online*. No modo *offline*, são utilizadas assinaturas digitalizadas a partir de dispositivos ópticos, como scanners e câmeras digitais, o que resulta em imagens estáticas da assinatura (PLAMONDON; SRIHARI, 2000). Este tipo de abordagem não é o mais eficiente para verificação de assinaturas devido ao fato de não possuir informações dinâmicas do movimento de escrita (NGUYEN; BLUMENSTEIN, 2010). Alguns trabalhos que utilizam essa abordagem para verificação de assinatura são (SCHAFFER; VIRIRI, 2009) e os trabalhos citados em Saikia e Sarma (2012).

Já no modo *online*, uma pessoa usa um elemento digitalizador (mesa digitalizadora, *tablet* ou celular) para converter a informação fornecida pela mão em assinatura, que na verdade é um banco de dados contendo diversas informações como: trajetória percorrida, velocidade da mão, pressão aplicada e inclinação da caneta (NGUYEN; BLUMENSTEIN, 2010). Essa abordagem é a que domina as pesquisas sobre verificação de assinatura atualmente, como os trabalhos de Canuto (2014) e Li et al. (2013). Isso acontece pois, devido à possibilidade de extrair mais informações da assinatura, os resultados encontrados usando essa abordagem têm sido melhores que os resultados encontrados usando abordagem *offline* (QIAO; LIU; TANG, 2007).

Ambas as abordagens têm seus prós e contras. Fechamentos de contrato, autenticações de responsabilidade, verificação de documentos, além de várias outras tarefas que exigem assinatura pessoal são geralmente realizadas em meio *offline*, ou seja, assinando em papel, e não há indícios de que isso vá mudar em um futuro próximo. A assinatura captada por meio *online* possui maior confiabilidade quanto à autenticação de documentos, no entanto, em vários casos é mais economicamente viável a utilização de sistemas *offline* que são suficientemente precisos para aplicações como endereçamento de cartas e autenticação de valor de cheques (NEL, 2005).

Na década de 2000 vários trabalhos focados em recuperar a trajetória da caneta em textos *offline* foram desenvolvidos, incluindo os dois que motivaram esta dissertação, sendo eles (NEL; PREEZ; HERBST, 2005) e (QIAO; LIU; TANG, 2007). Plamondon e Srihari em (PLAMONDON; SRIHARI, 2000) defendem a relevância deste tópico de pesquisa com a seguinte declaração:

“O sucesso dos sistemas *online* torna atraente considerar o desenvolvimento de sistemas *offline* que primeiro estimem a trajetória da escrita a partir dos dados *offline* e, em seguida, usem algoritmos de reconhecimento *online*. No entanto, a dificuldade de recriar os dados temporais levou a poucos desses sistemas de extração de características até agora.”(Tradução livre feita pelo autor)

O trabalho de Munich e Perona (2003) reforça que a trajetória da caneta em assinaturas contribui para a eficiência de um verificador *online*. De fato, no trabalho de Qiao, Liu e Tang (2007), que verifica autenticidade de assinaturas *offline* com ordem de escrita recuperada comparando-as com assinaturas *online* registradas, é obtida uma EER média de 7,4% na sua base de dados, o que, segundo o autor, encontra-se na mesma faixa de EER dos métodos puramente *online* e muito melhor que os métodos puramente *offline* testados em seu trabalho.

O processo de recuperação de trajetória, no entanto, não é um processo simples, vários problemas ocorrem quando se tenta determinar a ordem de escrita de um texto (ou assinatura). Esses desafios serão apresentados na seção 1.1.

1.1 Extração de trajetória e suas dificuldades

A extração da trajetória da caneta em textos¹ *offline* não é uma tarefa trivial. A largura do traço no papel, a quantidade de linhas sobrepostas e presentes em uma interseção, a variância no formato de texto de um mesmo autor, são apenas algumas das características que tornam essa tarefa bastante complexa. No caso deste trabalho, que trata de assinaturas, o nível de complexidade pode aumentar muito, dada a característica pessoal deste tipo de texto. Um bom exemplo de uma assinatura bastante complexa pode ser visto na Figura 1(a) em que, à primeira vista, é praticamente impossível retrair o caminho que a caneta realizou sem conhecimento prévio.

Uma necessidade primordial do nosso trabalho é encontrar o ponto de início e fim do movimento da caneta. Estes pontos estão frequentemente escondidos dentro da imagem (especialmente no caso de assinaturas) e encontrá-los torna-se mais desafiador

¹ Por conveniência, quando falarmos em “texto” estaremos nos referenciando a textos e assinaturas manuscritas.

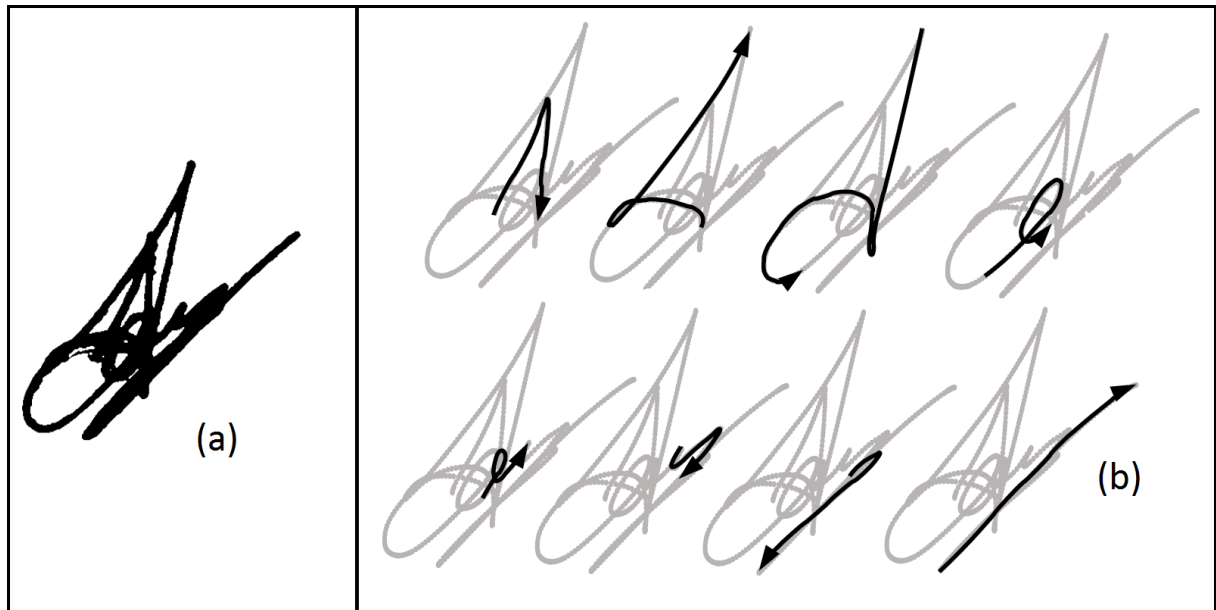


Figura 1 – (a) Exemplo de assinatura com formato complexo. (b) Exemplar *online* da mesma assinatura em (a). Adaptado de [Nel \(2005\)](#).

se a assinatura é constituída de vários *strokes*², já que um novo par (posição inicial e final) é gerado a cada *stroke*. Conhecer informação de pressão aplicada à caneta é vital para identificar esses pontos, sendo que os pontos onde a pressão é zero podem indicar movimento de caneta fora do papel, tornando a localização dos eventos de tocar e levantar de caneta trivial. Esta informação, no entanto, não está diretamente presente na assinatura *offline*, mas alguns trabalhos como ([WIROTIUS; SEROPIAN; VINCENT, 2003](#)) e ([PERVOUCHINE; LEEDHAM; MELIKHOV, 2005](#)) tentam extrair esta informação baseando-se no nível de cinza das imagens. Nel em ([NEL, 2005](#)) afirma que mesmo tendo essa estimativa, não é possível confiar nela quando o texto possui muitos pontos de cruzamento (interseções). Por causa destas dificuldades, muitos estudos tratam apenas de textos com um *stroke* ([NEL; PREEZ; HERBST, 2005; KATO; YASUHARA, 2000](#)).

Em assinaturas, frequentemente são encontradas regiões em que várias linhas se cruzam. Essas regiões são um problema quando se está recuperando o caminho percorrido pela caneta, já que ao chegar nelas, se torna ambígua a escolha de qual linha se deve optar para seguir o movimento. Uma possibilidade é assumir que a direção do movimento se mantém ao entrar e sair de uma interseção ([NEL, 2005](#)). A escolha entre as diferentes possibilidades de caminhos na interseção geralmente é baseada em algum critério de suavidade para realizar decisões locais, como em ([LEE; PAN, 1992; BOCCIGNONE et al., 1993; LALLICAN; VIARD-GAUDIN, 1997; CHANG; YAN, 1999; HUANG; RONG; BIAN, 1995](#)). Esses critérios, no entanto, nem sempre são suficientes para encontrar o caminho correto quando o número de cruzamentos na interseção é muito grande, por causa

² Consideramos *stroke* a parte da escrita entre um evento de tocar de caneta e um evento de levantar de caneta.

disso, trabalhos como Huang e Yasuhara (1995) e Kato e Yasuhara (2000) limitam em dois o número de cruzamentos possíveis em uma interseção. Pode ser visto na Figura 1(a) que esta restrição não é necessariamente válida em casos que tratam de assinaturas. Para sanar esta deficiência nos métodos que usam decisões locais, vários estudos introduzem informação global, modelando o sistema como um problema de teoria de grafos, como em (QIAO; LIU; TANG, 2007; JÄGER, 1998; KATO; YASUHARA, 2000; LALLICAN; VIARD-GAUDIN; KNERR, 2000).

Por via de regra, os estudos citados anteriormente utilizam apenas informação estática do texto (figura digitalizada da assinatura). Outra abordagem para isso é obter o referencial dinâmico correspondente usando uma mesa digitalizadora (GUO; DOERMANN; ROSENFELD, 2001; LAU; YUEN; TANG, 2003). A ideia é comparar o texto estático (*offline*) com o exemplar dinâmico armazenado, e usar os resultados dessa comparação para melhorar o desempenho da recuperação de trajetória. Essa abordagem simplifica o processo de obtenção dos pontos de início e fim de trajetória e também dá bons indícios de quantos *strokes* estão presentes no texto, já que o exemplar dinâmico possui essas informações na sua essência. A Figura 1(b) representa o exemplar *online* da Figura 1(a), nela pode ser visto uma quantidade muito maior de informações referentes ao caminho que a caneta percorreu para gerar a Figura 1(a), inclusive que assinatura possui apenas um *stroke*. Outra vantagem desta abordagem é a identificação de pontos de retorno, onde a caneta para e então muda de direção, informação essa que não existe nos exemplares *offline* (NEL, 2005). Sabendo das vantagens existentes ao utilizar amostras *online* para recuperar trajetórias em textos *offline*, é necessário também discutir algumas características que causam problemas ao utilizar essa abordagem.

Bases de dados que possuem assinaturas no formato *online* em geral possuem várias amostras de uma mesma assinatura. Por causa da característica da escrita humana, variações geométricas³ são bastante presentes nessas amostras o que, em geral, é um problema ao se fazer a comparação dos exemplares com a assinatura *offline* (PLAMONDON; SRIHARI, 2000). Para contornar isso, Qiao, Liu e Tang (2007) compara todas as assinaturas *online* de um mesmo usuário em sua base de dados e encontra a amostra que retorna menor valor médio de uma distância elástica, usando o método *Dynamic Time Warping* (DTW) para isso, essa amostra é então utilizada como referência para treinar o seu sistema. Já Nel, Preez e Herbst (2005) aplicam seu algoritmo em todas as amostras de mesma assinatura da sua base, as saídas deste processo são então comparadas ao resultado do algoritmo usando o referencial dinâmico⁴ da assinatura *offline* como entrada, a que obtiver maior verossimilhança é considerada a correta. Outro problema causado pelas variações geométricas é que nem sempre o exemplar dinâmico pode ser usado para recuperar a

³ Variações geométricas são variações de forma da escrita, por exemplo, posição, orientação, tamanho e inclinação.

⁴ Referencial dinâmico é o exemplar *online* captado no mesmo instante que o exemplar *offline* é gravado.

assinatura completa, visto isso, trabalhos como [Guo, Doermann e Rosenfeld \(2001\)](#) e [Lau, Yuen e Tang \(2003\)](#) utilizam os exemplares como auxílio apenas em decisões locais nas interseções da assinatura *offline*.

De modo geral, existem dois procedimentos principais no processo de recuperação de trajetória: examinação local e reconstrução global ([NGUYEN; BLUMENSTEIN, 2010](#)). A examinação local geralmente inclui análise de regiões como pontos de bifurcação, zonas ambíguas, linhas traçadas mais de uma vez e regiões de terminação. O processo de reconstrução global então analisa essas informações enquanto reconstrói a trajetória completa. Um esquema com os principais componentes de um sistema de recuperação de trajetória pode ser visto na Figura 2. Nesta figura é possível ver que esse tipo de sistema, em geral, é dividido em três passos: (i) pré-processamento, em que a imagem da assinatura *offline* e/ou a assinatura *online* são processadas de modo a simplificar a extração de suas características; (ii) examinação local, em que a assinatura *offline* tem características como *loops* e regiões ambíguas extraídas; e (iii) reconstrução global, em que as informações obtidas nos passos anteriores são utilizados para recuperar a trajetória global da assinatura *offline*.

Conhecendo os desafios e passos para o desenvolvimento de um sistema de recuperação de trajetória de caneta em assinaturas *offline* fica claro que a área é bastante promissora para pesquisas, esse foi um dos principais motivos para a escolha do tema que será melhor explorado nos capítulos a seguir.

1.2 Objetivos

Este trabalho tem como objetivo geral estudar e implementar métodos de recuperação de trajetória de caneta em assinaturas *offline* utilizando informação de assinaturas *online* previamente registradas como referência. Como objetivos específicos temos:

- Realizar uma ampla revisão bibliográfica sobre os trabalhos na área de recuperação de trajetória de ponta de caneta;
- Estudar e implementar técnicas de extração do esqueleto das imagens das assinaturas *offline*;
- Investigar e reproduzir métodos de extração de características de assinaturas *offline* para aproxima-las de assinaturas *online*;
- Implementar um sistema que recupere a trajetória da caneta sendo auxiliado pela informação contida em assinaturas *online* registradas previamente;
- Comparar os resultados com trabalhos anteriores.

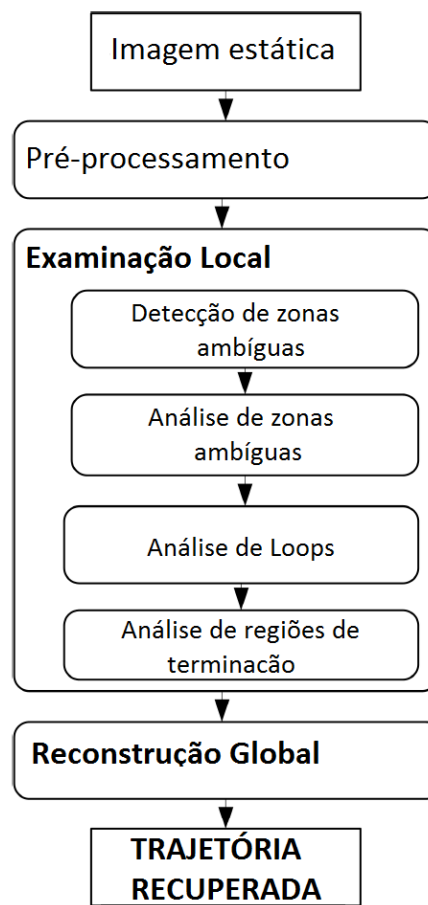


Figura 2 – Esquema de um sistema de recuperação de trajetória padrão. Adaptado de [Nguyen e Blumenstein \(2010\)](#).

1.3 Organização

Os capítulos seguintes deste trabalho estão organizados da seguinte maneira: No Capítulo 2 é abordado o processamento da assinatura *offline* feito a partir da extração do seu esqueleto, no início do capítulo é apresentada uma revisão do tema esqueletização e algumas das suas subáreas, o capítulo se encerra com a descrição do algoritmo implementado para este trabalho. No Capítulo 3 é explorada a extração de características das assinaturas *online* e *offline*, sendo que no início do capítulo é feita uma revisão de alguns trabalhos sobre o tema e no final são descritos os métodos implementados para a dissertação. O Capítulo 4 é focado na reconstrução da assinatura utilizando as informação obtidas nos capítulos anteriores. Seguindo a organização dos capítulos 2 e 3, o capítulo 4 também possui no seu início um conjunto de trabalhos desenvolvidos sobre seu tema e no final é apresentado o método desenvolvido. Já o Capítulo 5 contém as descrições dos experimentos realizados e seus resultados, cada um com sua respectiva discussão. Por fim, o Capítulo 6 contém as conclusões sobre o trabalho realizado.

2 Esqueletização

Ao gerar uma assinatura, uma pessoa transfere informações do cérebro para os músculos do braço, que por sua vez movimentam uma caneta sobre uma superfície produzindo uma sequência de traços que ao final do processo representarão a assinatura (NEL; PREEZ; HERBST, 2005). Os traços resultantes deste processo possuem largura que dependem do instrumento utilizado para produzi-los, por exemplo uma assinatura escrita com uma caneta com ponta de 1 milímetro terá traços com largura média de 1 milímetro. Quando a assinatura é captada por meios *online*, no entanto, essa variabilidade não acontece. Ao passar sobre o elemento digitalizador as posições nas quais a ponta da caneta esteve são amostradas (tipicamente - mas não necessariamente - a intervalos de tempo constantes) e unidas sequencialmente. O resultado disso é uma assinatura com formato de “linha”. Assinaturas *offline*, por sua vez, possuem formato de “fita”, ou seja, finas e alongadas. Esse tipo de estrutura é bastante comum em processamento de imagem sendo que Zou e Yan (2001) afirmam que o esqueleto dessas imagens representa de forma satisfatória a geometria da estrutura. Sendo o esqueleto uma estrutura que se aproxima de um formato de “linha” temos uma boa solução para aproximar assinaturas *offline* de assinaturas *online*.

Extrair o esqueleto de uma imagem significa reduzir a imagem a um grafo conectado ou uma estrutura com apenas um *pixel* de largura (GONZALEZ; WOODS, 2007). Uma boa revisão bibliográfica sobre esqueletização em geral pode ser vista em (SAHA; BORGEFORS; BAJA, 2016). As seções seguintes servirão para apresentar diferentes algoritmos de esqueletização e alguns trabalhos que se destacam usando cada um deles.

2.1 Esqueletização Clássica

Os algoritmos de esqueletização a princípio são algoritmos de afinamento de imagem até que a estrutura tenha apenas um *pixel* de largura. Gonzalez e Woods (2007) apresentam a forma que, por conveniência¹, iremos considerar como algoritmo clássico de esqueletização. Nele uma imagem preta e branca (binária) tem seus *pixels* de borda excluídos até que a estrutura possua apenas um *pixel* de largura. Para isso os *pixels* de borda são marcados de acordo com quatro condições em dois passos. Considere a Figura 3 como referência para a organização dos *pixels* p_1, p_2, \dots, p_9 para o algoritmo, considere também que p_1 é o *pixels* de borda que está sendo processado, assim, temos que no primeiro passo as seguintes condições são testadas:

¹ Esse é o algoritmo selecionado para ser o algoritmo clássico pois está descrito em um dos livros mais utilizados sobre processamento de imagem que existe, facilitando a busca pelo texto original.

- (a) $2 \leq N(p_1) \leq 6$, sendo que $N(p_1)$ é o número de *pixels* brancos² na vizinhança-8³ de p_1 ;
- (b) $T(p_1) = 1$, sendo que $T(p_1)$ é o número de transições de *pixels* pretos para *pixels* brancos seguindo a sequência $p_2, p_3, \dots, p_8, p_9$;
- (c) $p_2 \cdot p_4 \cdot p_6 = 0$;
- (d) $p_4 \cdot p_6 \cdot p_8 = 0$;

No segundo passo as condições a e b se mantêm sendo que as outras duas ficam da seguinte forma:

- (c') $p_2 \cdot p_4 \cdot p_8 = 0$;
- (d') $p_2 \cdot p_6 \cdot p_8 = 0$;

Caso um *pixel* de borda não supra as quatro condições ele é marcado para ser excluído ao final do passo atual. O sistema então, repete os passos um e dois seguidamente até que não haja mais exclusão de *pixels*. Na Figura 4 pode ser visto o resultado deste processo.

p_9	p_2	p_3
p_8	p_1	p_4
p_7	p_6	p_5

Figura 3 – Organização dos *pixels* para o algoritmo de esqueletização (GONZALEZ; WOODS, 2007).

Com o esqueleto computado, iremos introduzir uma notação usada em vários trabalhos que utilizam o processo de esqueletização: grau dos *pixels* e suas nomenclaturas (BOCCIGNONE et al., 1993; LEE; PAN, 1992; QIAO; LIU; TANG, 2007; HUANG; YASUHARA, 1995; ZOU; YAN, 2001; NEL; PREEZ; HERBST, 2005). Considere a célula p_1 na Figura 3 como um *pixel* do esqueleto, o grau de p_1 será o número *pixels* do esqueleto

² Consideramos, para o processamento da imagem, *pixels* brancos com nível lógico 1 e *pixels* pretos com nível lógico 0. As imagens preto e branco presentes neste trabalho, no entanto, estão representadas com *pixels* brancos sendo nível lógico 0 e *pixels* pretos com nível lógico 1, isso é feito para melhor visualização das imagens.

³ Vizinhança-8 são os oito *pixels* que estão ao redor de p_1 .

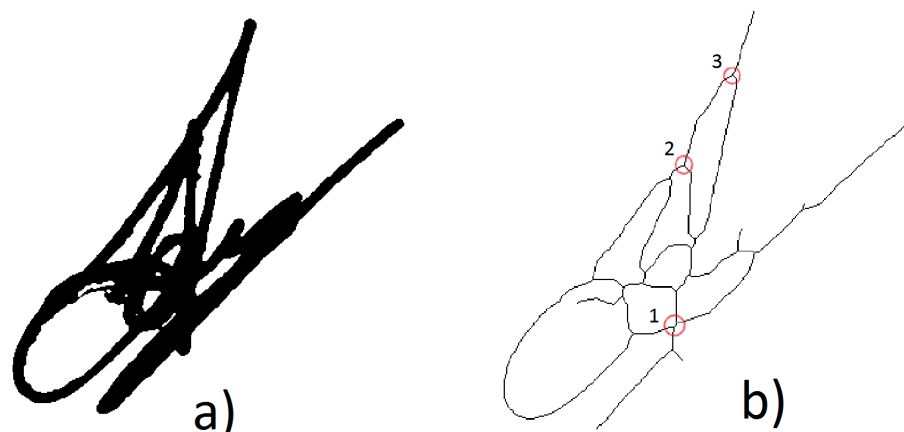


Figura 4 – a) Assinatura original em preto e branco (formato de “fita”). b) Resultado do processo de esqueletização usando o algoritmo clássico. Os círculos vermelhos apontam alguns artefatos resultantes deste processo (formato de “linha”).

presentes na vizinhança-8 dele, ou seja, se p_1 possui apenas um vizinho ele é de grau 1, se possui dois vizinhos ele é de grau 2, e assim sucessivamente. Como nomenclatura temos várias classificações para os mesmos graus, mas em maior frequência temos que *pixels* de grau 1 são chamados de *pixels* (ou pontos) de terminação, os de grau 2 são chamados de *pixels* normais e de grau maior que 2 são chamados de *pixels* de junção. Há também os *pixels* de grau 0, chamados de *pixels* isolados, mas esses são menos frequentes na literatura, pois geralmente são excluídos do esqueleto.

Como pode ser visto, o algoritmo é pouco complexo e de fácil implementação, porém os círculos vermelhos na Figura 4(b) indicam um problema decorrente dessa abordagem. O algoritmo clássico de esqueletização é bastante sensível a ruído e largura da linha, gerando artefatos indesejados em regiões de bifurcação e terminação (LEE; PAN, 1992). Esses artefatos são um problema sem solução definida na literatura, no entanto vários trabalhos propõem maneiras de contornar esses defeitos. Lee e Pan (1992) consideram artefatos de bifurcação (círculo 1 na Figura 4(b)) o maior problema referente à estimação de caminho percorrido ao escrever. Para tratá-los eles encontram o ponto médio do segmento interno e interpola com os outros pontos pertencentes à bifurcação. Artefatos de alongamento (número 3 na Figura 4(b)) não são processados em seu trabalho pois o autor afirma que não causam problemas sérios no processo de recuperação da trajetória.

Para obter o esqueleto primário, Boccignone et al. (1993) usam um algoritmo de Transformada de Eixo Médio (*Medial Axis Transform*, MAT) que é uma abordagem semelhante ao algoritmo clássico aqui citado, gerando os mesmos tipos de artefatos. Para tratá-los é realizada uma MAT com aproximações poligonais da imagem que os autores chamaram de PMAT e, em seguida, a corrige utilizando critérios de continuidade, processo esse batizado de CPMAT. Na Figura 5 é possível ver as etapas desse processo. Uma boa revisão sobre esse tipo de esqueletização pode ser vista em (LAM; LEE; SUEN, 1992).

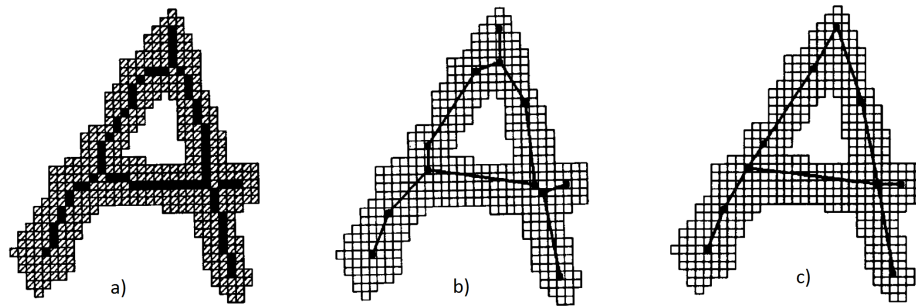


Figura 5 – a) Esqueleto primário sobre o texto original obtido via MAT. b) Esqueleto com PMAT aplicada. c) Aplicação da CPMAT sobre o esqueleto. Adaptado de (BOCCIGNONE et al., 1993)

Para tentar eliminar os artefatos gerados pelos algoritmos de afinamento, vários autores desenvolveram algoritmos de pseudo-esqueletização que em sua essência não possuem essas características indesejadas. Esses algoritmos serão tratados na seção 2.2 e na seção 2.3 será descrito o algoritmo de esqueletização utilizado neste trabalho.

2.2 Pseudo-Esqueletização

Como foi visto na seção 2.1, os algoritmos de esqueletização que se baseiam em algoritmos de afinamento, como os propostos em Gonzalez e Woods (2007), geram características indesejadas que fazem com que o esqueleto não represente bem certas regiões do texto original. Para solucionar isso, alguns trabalhos foram desenvolvidos para encontrar o esqueleto de imagens sem usar algoritmos de afinamento diretamente. Nesta seção serão citados alguns trabalhos relevantes sobre esses algoritmos que são chamados de algoritmos de pseudo-esqueletização e são definidos por Steinherz, Intrator e Rivlin (2000) como imagens com um *pixel* de largura que satisfazem as seguintes condições:

- Cada *loop* da imagem original é associado a um *loop* de um *pixel* de largura com o mesmo formato, tamanho proporcional e aproximadamente no mesmo local;
- O final de cada traço é associado a um *pixel* de terminação ⁴;
- Cada interseção na imagem original é associada a um *pixel* de junção ⁵;
- O tamanho e curvatura dos traços é preservada.

Kégl e Krzyzak (2002) propuseram um método de esqueletização linear por partes usando Curvas Principais, que são curvas suaves que passam pelo meio de uma nuvem de pontos. Para isso ele utiliza quatro passos, sendo eles: inicialização, ajuste e suavização,

⁴ Ver seção 2.1

⁵ Ver seção 2.1

reconstrução e ajuste e suavização novamente. O passo de inicialização é feito usando um algoritmo de afinamento como os da seção 2.1, esse esqueleto inicial é convertido em um grafo G . O segundo passo, ajuste e suavização, é feito realocando vértice e arestas de G baseado nas suas posições em relação aos *pixels* da imagem original. O passo de reconstrução elimina ou retifica imperfeições como ramos e *loops* curtos e interseções mal formadas. E por último, o passo de ajuste e suavização, segundo os autores, torna o esqueleto final mais próximo do movimento real feito ao gerar a imagem. Na Figura 6 são ilustradas as etapas do algoritmo. Esse método, no entanto ainda usa um algoritmo de afinamento, mesmo que apenas para inicialização.

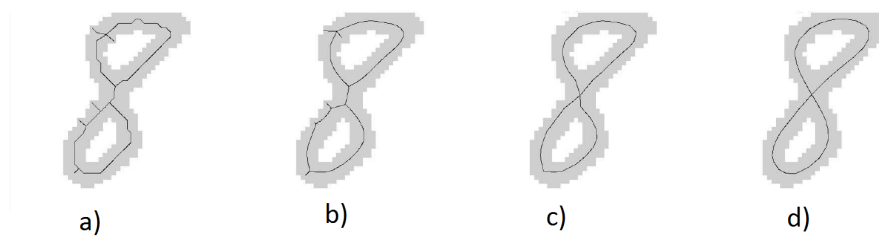


Figura 6 – a) Esqueleto de inicialização feito usando o método clássico. b) Esqueleto inicial após a etapa de Ajuste e Suavização. c) Aplicação da etapa de Reconstrução. d) Ajustes e Suavizações finais. Adaptado de (KÉGL; KRZYŻAK, 2002)

Pervouchine, Leedham e Melikhov (2005) por outro lado, não utilizam algoritmo de afinamento no seu trabalho. Sua abordagem utiliza imagens em tons de cinza para extrair esqueletos. Segundo os autores, não utilizar imagens em preto e branco permite identificar características que, caso contrário, seriam perdidas como pequenos *loops* e pequenas curvas. Para extrair o esqueleto dessas imagens o algoritmo segmenta a imagem em regiões que podem ser aproximadas por retângulos proporcionais à largura do traço escrito. Essas regiões têm seus centros marcados como nós que serão utilizados para ancorar as *Splines* responsáveis por unir esses nós, gerando o esqueleto final.

Dawoud e Kamel (2003) também encontram o esqueleto de imagens em tons de cinza. Em sua abordagem o esqueleto é extraído após limiarizações sucessivas da imagem original extraindo pontos que alongam o esqueleto e excluindo os que o engrossam. A limiarização é feita diversas vezes iniciando do menor nível de cinza possível (mais próximo do preto) até o maior possível (branco) e o esqueleto vai se formando a partir de cada iteração como pode ser visto na Figura 7.

Steinherz, Intrator e Rivlin (2000) computam o pseudo-esqueleto de palavras escritas de maneira cursiva baseando-se em dois elementos denominados *axis* e *tarsi*. O *axis* é o caminho mais curto, interno aos contornos da imagem, entre os *pixels* mais a esquerda da imagem e os *pixels* mais a direita. Já o *tarsi* é o conjunto de *pixels* de terminação (citados na seção 2.1) com maior distância do *axis*, porém ainda conectados, e

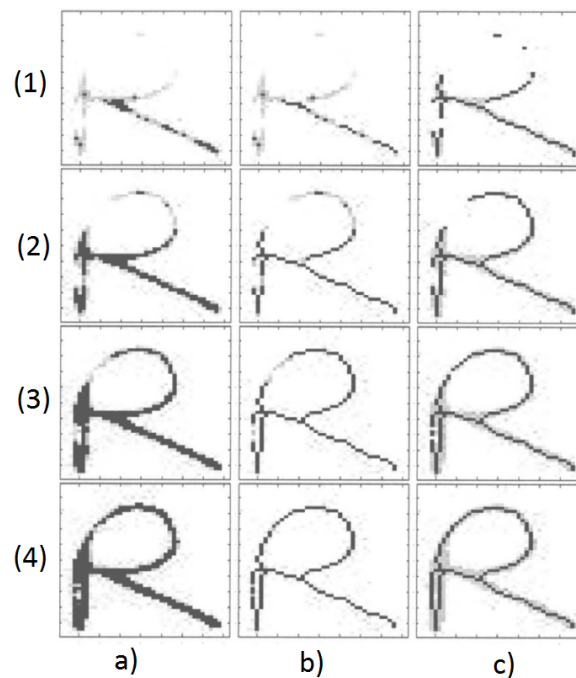


Figura 7 – Método de esqueletização em [Dawoud e Kamel \(2003\)](#), as linhas representam as iterações do algoritmo e as colunas os passos em cada iteração. a) *Pixels* em preto são os selecionados na iteração (limiarização) anterior, *pixels* em cinza são os selecionados na iteração atual. b) Tratamento dos *pixels* do esqueleto resultante da iteração anterior com os *pixels* selecionados na iteração atual. c) Esqueleto resultante na iteração atual.

loops extraídos a partir das regiões internas do contorno. Ao final, o esqueleto resultante é suavizado para melhor representação do texto.

A seção a seguir irá descrever o algoritmo selecionado para ser utilizado neste trabalho.

2.3 Algoritmo Implementado

Para este trabalho o algoritmo escolhido para extrair o esqueleto das assinaturas *offline* foi o desenvolvido por [Zou e Yan \(2001\)](#). Este algoritmo foi escolhido pois, segundo o autor, produz um esqueleto mais próximo da percepção humana de como o esqueleto deveria ser desenhado. Além disso, este método é caracterizado por um dos trabalhos referência para esta dissertação ([NEL; PREEZ; HERBST, 2005](#)) como “bastante adequado” para extrair esqueletos de assinaturas. Sabendo disso, esta seção irá descrever os passos do algoritmo implementado.

Primeiramente a imagem tem suas bordas extraídas, usando um filtro de bordas de Canny, (Figura 8(b)) e organizadas em um grafo (G) em que os vértices são os pixels e as arestas são as conexões bidirecionais entre pixels vizinhos. O interior do contorno da

assinatura é então segmentado em triângulos por via da triangulação de Delaunay limitada pelo grafo G (CHEW, 1989), como pode ser visto na Figura 8(c).

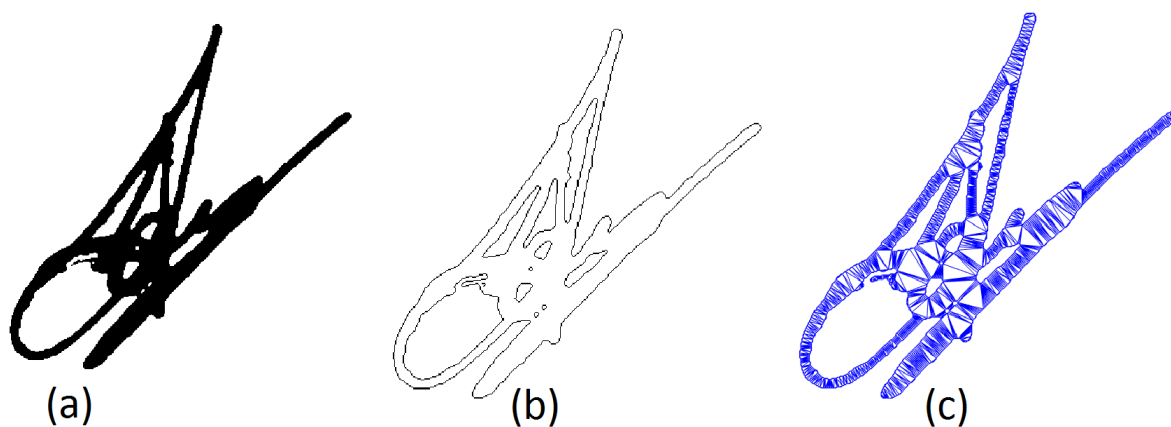


Figura 8 – (a) Assinatura binarizada. (b) Extração das bordas da assinatura. (c) Resultado da Triangulação Limitada de Delaunay.

A partir da assinatura segmentada é possível construir o esqueleto primário. Para isso é feita a união dos pontos característicos dos quatro tipos de triângulos que surgem ao realizar a triangulação. Consideramos um triângulo isolado (TI) aquele que não possui nenhuma de suas arestas no interior do contorno. Já os triângulos de terminação (TT) são aqueles que possuem apenas um de seus lados no interior do contorno. Um triângulo normal (TN), por outro lado, é aquele que possui dois dos seus lados no interior do contorno, estes são os triângulos que aparecem mais frequentemente. Por fim, um triângulo de junção (TJ) é aquele que possui todos os seus lados no interior do contorno. O ponto característico de todos os triângulos são seus centroides, com exceção dos TN que possui como ponto característico o ponto médio entre os pontos médios de suas arestas internas. Na Figura 9(a) é ilustrada a identificação dos triângulos encontrados e seus pontos característicos. Triângulos isolados são bastante raros de encontrar na fase inicial de segmentação por isso, nesta base de dados, nenhum foi encontrado.

Como pode ser visto na Figura 9(b), o esqueleto primário possui vários ramos que não correspondem necessariamente a movimento de caneta. Segundo Zou e Yan (2001), vários dos triângulos responsáveis pelos pontos do esqueleto primário não correspondem à percepção humana, por isso eles devem ser reprocessados. O processamento dos triângulos de terminação será detalhado na seção 2.3.1 e o processamento dos triângulos de junção na seção 2.3.2. Não é necessário processar triângulos normais ou isolados. A partir deste ponto, o processamento dos triângulos fará com que alguns deles sejam unidos com seus vizinhos, fazendo-os deixar de ser triângulos, por esse motivo utilizaremos o termo polígonos para nos referir a todas as partições da assinatura. O tipo de polígono é equivalente ao tipo de triângulo em relação às arestas internas (TI, TT, TN e TJ), ou seja, polígonos isolados não

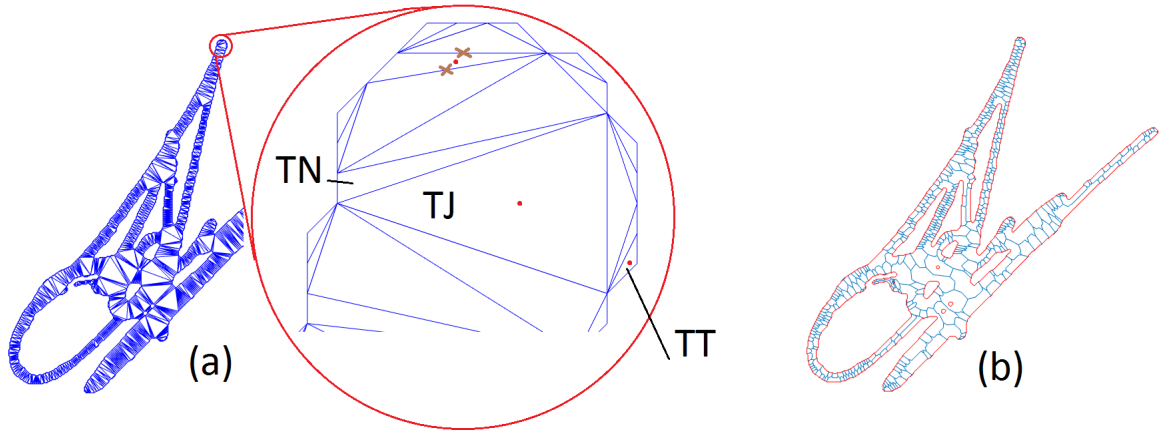


Figura 9 – a) Ampliação de um pedaço da assinatura segmentada e identificação dos tipos de triângulos formados. Em vermelho os pontos característicos de cada um dos triângulos. Os “x” representam os pontos médios das arestas internas do TN que geram seu ponto característico. b) Esqueleto primário da assinatura.

possuem arestas no interior do contorno, polígonos de terminação possuem apenas uma aresta interna ao contorno, polígonos normais possuem duas arestas internas e polígonos de junção possuem mais que duas arestas internas.

2.3.1 Processamento de Polígonos de Terminação

Cada polígono de terminação possui um círculo característico C com centro localizado no ponto médio da aresta interna do polígono e diâmetro $\sqrt{2}w$, em que w é o comprimento da aresta mais longa do polígono vizinho. O polígono de terminação é considerado instável se nenhum de seus vértices estiver no exterior o círculo C . Sendo instável, o polígono é unido ao polígono vizinho até que este se torne estável. Na Figura 10 é ilustrado esse processo.

Esse processo é feito até que não existam polígonos de terminação instáveis restantes.

2.3.2 Processamento de Polígonos de Junção

Os polígonos de junção têm característica de ser encontro de três ou mais retas referentes ao movimento da ponta da caneta. Zou e Yan (2001) consideram que o ponto de encontro dessas retas, também chamado Ponto Característico (P), deve minimizar a distância entre elas. Para encontrar este ponto, considere

$$J(x,y) = \sum_{i=1}^n d_i^2(x,y), \quad (2.1)$$

em que $d_i(x,y)$ é a distância entre o ponto $P(x,y)$ e a reta r_i . O ponto P deve minimizar $J(x,y)$ em x e y , ou seja,

$$\begin{cases} \frac{\partial J(x,y)}{\partial x} = 0 \\ \frac{\partial J(x,y)}{\partial y} = 0 \end{cases} . \quad (2.2)$$

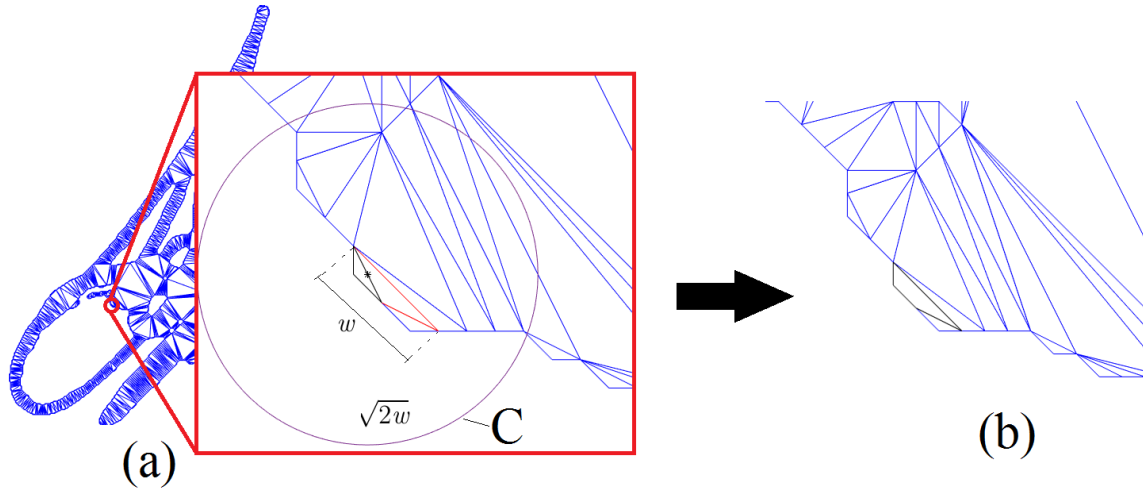


Figura 10 – (a) Polígono testado em preto e seu círculo característico em roxo. (b) União dos dois polígonos por não suprir a condição de estabilidade

Considere também que a distância entre um ponto e uma reta seja

$$d(x, y) = \frac{|ax + by + c|}{\sqrt{a^2 + b^2}}, \quad (2.3)$$

em que a , b e c são os parâmetros da reta e x e y as coordenadas do ponto. Substituindo (2.3) em (2.1) e aplicando em (2.2), temos

$$\begin{cases} \sum_{i=1}^n \frac{2a_i^2x + 2a_ib_iy + 2a_ic_i}{a_i^2 + b_i^2} = 0 \\ \sum_{i=1}^n \frac{2a_ib_ix + 2b_i^2y + 2b_ic_i}{a_i^2 + b_i^2} = 0 \end{cases} \quad (2.4)$$

Assim, o ponto P é a solução da equação 2.4. Os parâmetros da reta r_i são encontrados a partir dos pontos característicos dos M vizinhos mais próximos na direção perpendicular à aresta i do polígono de junção, ou até encontrar um outro polígono de junção, como pode ser visto na Figura 11(a). O valor de M é um parâmetro de ajuste, Zou e Yan (2001) não definem um valor exato mas Nel (2005) utiliza 16 para essa variável, como a base de assinaturas utilizada neste trabalho é a mesma utilizada em Nel (2005), o valor de M utilizado aqui também é 16. Um polígono de junção é considerado instável quando o ponto característico P se encontra fora do polígono. Caso isso aconteça, o polígono será unido com os polígonos vizinhos na direção de P até que eles estejam no interior do polígono de junção. Na Figura 11(b) esse processo é ilustrado. Caso o ponto P esteja fora dos contornos da assinatura, os polígonos serão unidos até que esteja o mais próximo possível de P , em seguida o ponto característico é colocado no centroide do polígono. Caso existam dois polígonos de junção vizinhos, estes serão unidos e o ponto característico recalculado. Com os polígonos de terminação e junção processados é possível gerar o esqueleto final unindo os pontos característicos de polígonos vizinhos, formando um grafo referente ao esqueleto. Na Figura 12 é mostrado o resultado deste processo em formato de esqueleto final (Figura 12(a)) e em formato dos polígonos processados (Figura 12(b)).

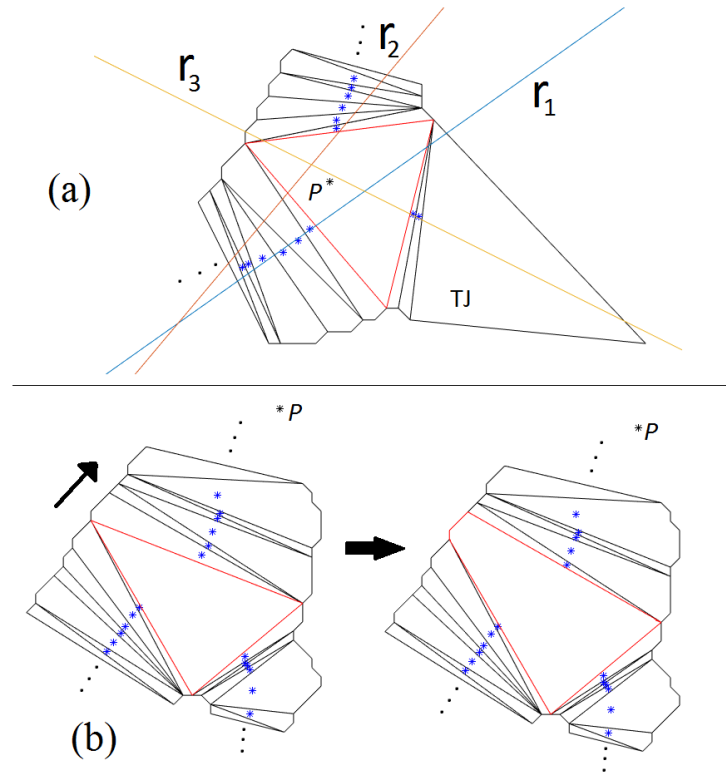


Figura 11 – (a) Em azul os pontos característicos dos polígonos vizinhos do polígono de junção testado (em vermelho) e em preto o ponto P que minimiza a distância entre as três retas. Só foram usados dois pontos para encontrar os parâmetro da reta r_3 pois o terceiro vizinho era um polígono de junção. (b) União dos polígonos na direção do ponto P

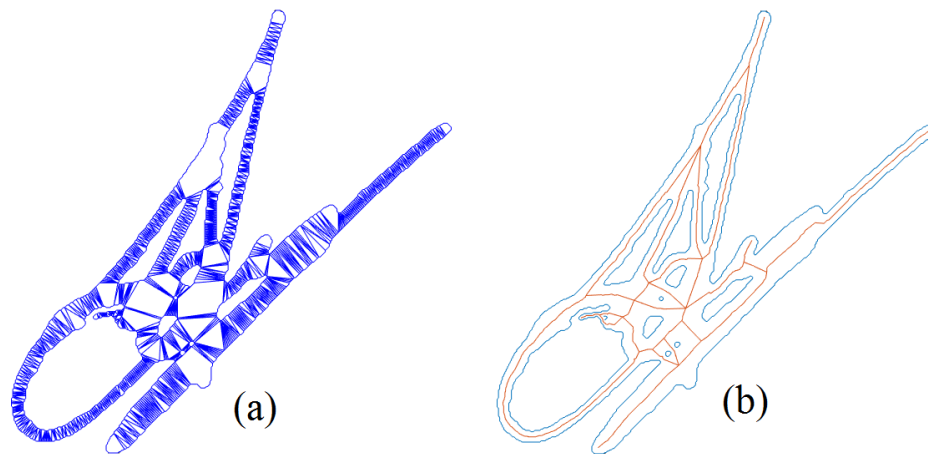


Figura 12 – (a) Resultado do processamento dos polígonos (b) Esqueleto Final

3 Extração de Características

Após extrair o esqueleto das imagens de assinaturas, temos uma representação simplificada da imagem original, mas com várias das suas características geométricas conservadas. Estas características, quando extraídas, dão informações essenciais de como a assinatura foi produzida, simplificando o processo de recuperação de trajetória. Além dessas características, o esqueleto também possui regiões em que o caminho percorrido pela caneta não é óbvio, como regiões de bifurcação e regiões mal formadas ou ruidosas. Este capítulo irá tratar de como extrair essas características e regiões sendo que, na seção 3.1 serão apresentados alguns trabalhos relevantes no tema e na seção 3.2 será apresentado e justificado o algoritmo implementado para este trabalho.

3.1 Trabalhos relevantes

O esqueleto em si é um conjunto de pontos que quando visto globalmente possui, até certo ponto, o formato da assinatura original. Quando se tenta ordenar os pontos do esqueleto para que estes reproduzam a trajetória percorrida pela ponta da caneta, no entanto, os pixels do esqueleto são em geral um aglomerado de pontos sem ordenação alguma. Vários trabalhos (LEE; PAN, 1992; QIAO; LIU; TANG, 2007; NEL; PREEZ; HERBST, 2005; HUANG; YASUHARA, 1995; QIAO; NISHIARA; YASUHARA, 2006) pré-ordenam o esqueleto modelando-o como um grafo $G(V, E)$ sendo V os vértices do grafo, representando os pontos do esqueleto e E as arestas representando as conexões (em geral bidirecionais) entre pontos vizinhos do esqueleto. Visto isso, trabalhos como (HUANG; RONG; BIAN, 1995; QIAO; NISHIARA; YASUHARA, 2006) usam a relação entre a largura média da assinatura original nos pontos do esqueleto que se conectem a mais de dois vizinhos e a largura média nas outras regiões como critério de identificação de regiões ambíguas.

Huang, Rong e Bian (1995) marcam os *pixels* do esqueleto em seu trabalho como confiável e não confiável baseado na largura do traço do texto original e no grau do *pixel* do esqueleto (se o *pixel* possui grau ≥ 3 , ele é considerado não confiável). O ponto do esqueleto que separa um ponto confiável de um não confiável é chamado ponto de conexão (*Conection Point*, C-point) e um grupo de pontos de conexão ligados por pontos não confiáveis é chamado de zona de conexão (*Conection Zone*, C-zone). A identificação dos pontos confiáveis e não confiáveis, C-points e C-zone pode ser vista na Figura 13. Os k pontos confiáveis mais próximos dos C-points são utilizados para calcular a direção D e a curvatura CV que serão usados como característica para serem comparadas dentro de um C-zone ligando ou não os C-points de acordo com um critério de continuidade definido

pela diferença entre essas características.

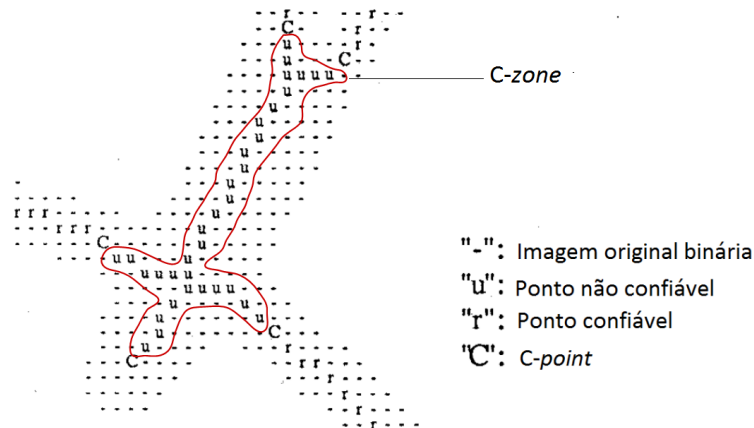


Figura 13 – Imagem adaptada da identificação das regiões ambíguas e pontos característicos no trabalho de Huang, Rong e Bian (1995).

No trabalho de Qiao, Nishiara e Yasuhara (2006) é utilizada uma abordagem semelhante ao trabalho de Huang, Rong e Bian (1995), nele é definida a Relação de Continuidade de Arestas (*Edge Continuity Relation*, ECR) para realizar decisões sobre as regiões ambíguas. Essas regiões são identificadas ao redor de *pixels* de grau ≥ 3 e grau $= 1$, como pode ser visto na Figura 14. Além disso, os nós também são classificados quanto à quantidade de “pernas” que ele possui, o nó 1 e o nó 2 na Figura 14, por exemplo, são de grau 6 e 4, respectivamente. Quando o nó é de grau 4, uma rede neural define a ordem em que as “pernas” do nó devem ser ligadas. Nesse trabalho também foi encontrado que 95,8% de todas as regiões ambíguas presentes nos experimentos eram de grau 4, e destes, 95,1% eram nós de cruzamento, ou seja, a presença deste tipo de região simplifica o processo de reordenamento por causa dessas relações.

Lee e Pan (1992), por outro lado, consideram os *pixels* do esqueleto da assinatura de grau 1 e de grau maior que 2 como pontos característicos. No trabalho, é considerado que o início ou final de um *stroke* está localizado, normalmente, sobre um *pixel* de grau 1. No entanto, quando isso não acontece a consideração é que o início (ou fim) do *stroke* está em um *pixel* de grau ímpar maior que 2. Também é considerado que *pixels* de grau par e maiores que 2 são pontos de cruzamento, estes *pixels* são então determinados como pontos intermediários quando o algoritmo de recuperação for aplicado, ou seja, serão traçados duas vezes. Os pontos característicos encontrados são então ligados para reordenar os pontos da assinatura utilizando um conjunto de regras heurísticas.

De maneira semelhante, Nel, Preez e Herbst (2005) marcam os *pixels* de grau 1 e grau maior que 2 do esqueleto em seu trabalho como regiões ambíguas. No entanto, além dessa marcação, o sistema é modelado como um grafo que é usado para construir um Modelo Escondido de Markov (*Hidden Markov Model*, HMM). Para adicionar um efeito de

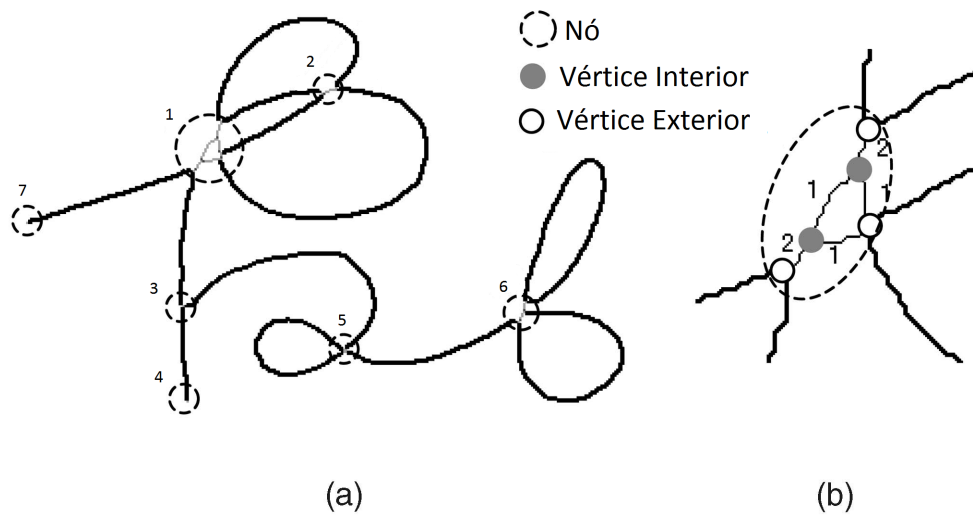


Figura 14 – (a) Esqueleto de uma palavra escrita a mão com os nós marcados de 1 a 7. (b) Ampliação do nó 1. Adaptado de (QIAO; NISHIARA; YASUHARA, 2006)

contexto ao processo de reconstrução, o modelo de Markov usado é um de segunda ordem, ou seja, que considera dois momentos anteriores ao estado atual. As regiões ambíguas encontradas são então utilizadas para alterar as probabilidades de transição entre estados utilizando regras específicas para cada uma delas. Na Figura 15 pode ser vista uma ilustração da modelagem do esqueleto em um HMM.

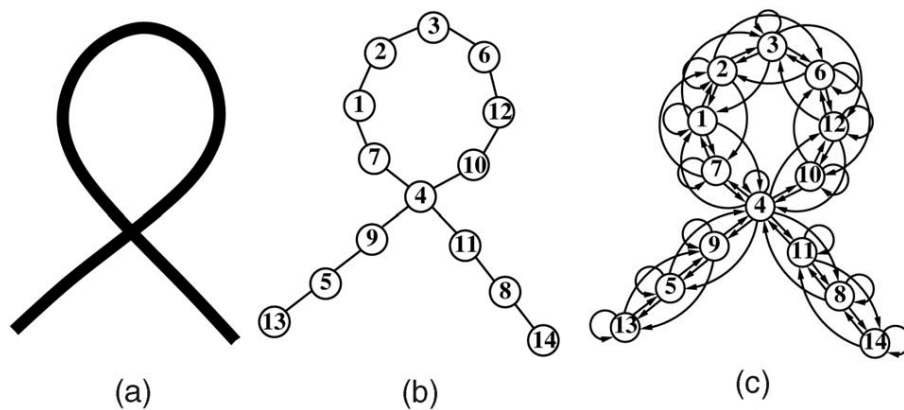


Figura 15 – Ilustração da modelagem de um esqueleto em um HMM. a) O esqueleto original representado de maneira contínua, por critério estético. b) Os *pixels* do esqueleto organizados em um grafo ligando *pixels* vizinhos. c) Esqueleto modelado como um HMM, as setas representam as possíveis transições que podem ocorrer. Adaptado de (NEL; PREEZ; HERBST, 2005).

Plamondon e Privitera (1999) diferente da maioria dos trabalhos citados até agora, não utilizam o esqueleto como fonte das características a serem extraídas. Em seu trabalho, os contornos do texto *offline* binarizado são extraídos e é a partir deles que as características

são encontradas. Para isso, são calculados os pontos de curvatura máxima nas regiões de contorno e estes pontos são considerados como as características principais do manuscrito. As regiões de máxima curvatura são então marcadas de acordo com a Figura 16 sendo: a) Alta curvatura no meio de um traço; b) Pontos de contato inicial ou final com o papel; c) Pontos de alta concavidade gerados por cruzamento de traços; d) Regiões que dois traços se unem para formar apenas um; e) Pontos de alta curvatura gerados por contato accidental entre traços de movimentos diferentes. Com os pontos classificados, a imagem é digitalizada usando uma janela de tamanho fixo e sempre que um agrupamento de pontos de alta curvatura for encontrado no seu interior, essa região é marcada como ambígua e é tratada usando critérios de continuidade.

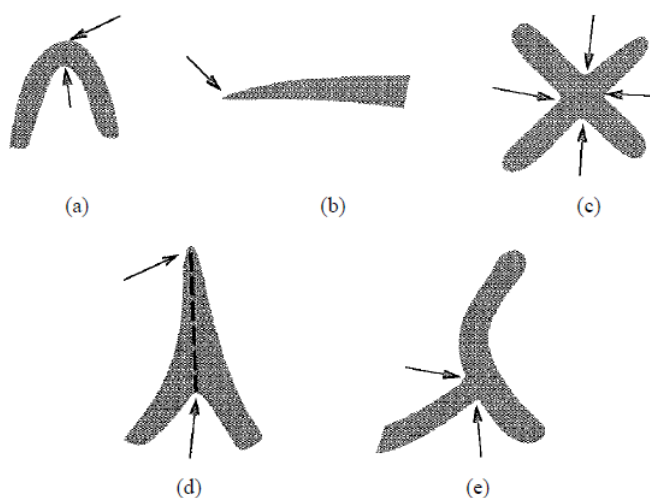


Figura 16 – Classificação das regiões de curvatura máxima segundo [Plamondon e Privitera \(1999\)](#).

De forma semelhante [Baati et al. \(2005\)](#) utilizam os pontos de máxima curvatura no contorno para encontrar regiões ambíguas. Inicialmente esses pontos são utilizados para encontrar os pontos dominantes do contorno (início e fim de traço, cruzamentos, bifurcações, etc). Após isso, é feita uma varredura da imagem original do texto utilizando uma janela de largura igual a três vezes a largura do traço. Quando essa janela passa sobre uma região que possui três ou quatro áreas em branco separadas, essas regiões são classificadas como bifurcação ou cruzamento, respectivamente. Uma ilustração dessas regiões pode ser vista na Figura 17. Após esse processo, são encontrados os pontos de máxima curvatura no interior das regiões ambíguas com o intuito de demarcar a área em que deve ser tomada decisão de que direção seguir, como pode ser visto na Figura 17(c).

3.2 Método Proposto

Como foi visto na seção anterior, vários trabalhos utilizam o grau dos *pixels* como característica específica para extração de informações do texto/assinatura *offline*. Neste

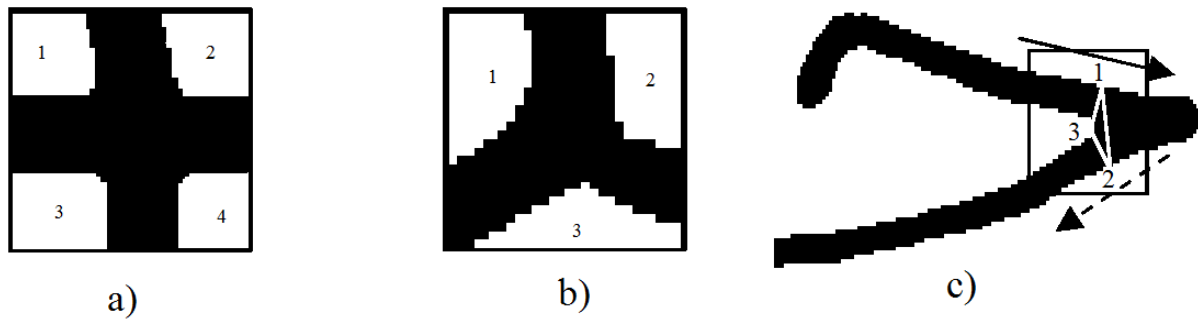


Figura 17 – Identificação das regiões ambíguas. a) Região de cruzamento. b) Região de bifurcação. c) Pontos de máxima curvatura nas regiões ambíguas gerando uma área de decisão. Adaptado de (BAATI et al., 2005).

trabalho não será diferente. Aqui, no entanto, ao invés de selecionar as regiões ambíguas do esqueleto para demarcar pontos de referência, o que foi extraído foram as regiões sem ambiguidade. Essa diferença no processo de extração faz com que seja criada uma pré-ordenação dos segmentos do esqueleto, sendo que, como o objetivo deste trabalho é recuperar a trajetória sobre esse esqueleto, essa pré-ordenação gera uma simplificação para este processo. Estas regiões, chamadas de segmentos do esqueleto conectados sem ambiguidade (*Unambiguously Connected Skeleton Segments*, UCSS), serão utilizados como caminhos possíveis que a ponta da caneta percorreu ao escrever a assinatura e serão ligados a partir de um critério de continuidade que será descrito no capítulo 4. Nesta seção, no entanto, será descrito como os UCSS são extraídos e sua aplicação em um trabalho desenvolvido durante a dissertação.

3.2.1 Definição de segmentos conectados sem ambiguidades - UCSS

Sabendo que o objetivo deste trabalho é recuperar o trajeto que a ponta da caneta fez ao escrever uma assinatura *offline*, sabendo também que isso será feito usando uma assinatura *online* como referência para o algoritmo, foi pensado um modo de realizar algum tipo ordenação dos pontos do esqueleto antes que este fosse aplicado ao algoritmo de recuperação. Essa pseudo-ordenação foi feita utilizando os Segmentos Diretamente Conectados do Esqueleto, ou UCSS¹.

O resultado do algoritmo de esqueletização utilizado neste trabalho é um grafo com os pontos do esqueleto como vértices e suas ligações sendo as arestas. A partir disso os vértices presentes entre:

1. Dois pontos do esqueleto de grau maior que 2;
2. Um ponto de grau 1 e um ponto de grau maior que 2, ou;

¹ Nome em inglês utilizado no trabalho desenvolvido para publicação e como será chamado a partir de agora.

3. Dois pontos de grau 1,

são agrupados e ordenados nos dois sentidos possíveis, como pode ser visto na Figura 18. A ordenação nos dois sentidos acontece por duas razões: seja porque o sentido do movimento é ambíguo, seja porque é possível que a caneta tenha passado duas vezes (ida e volta) ao longo daquele segmento (o que seria percebido como apenas um segmento offline).

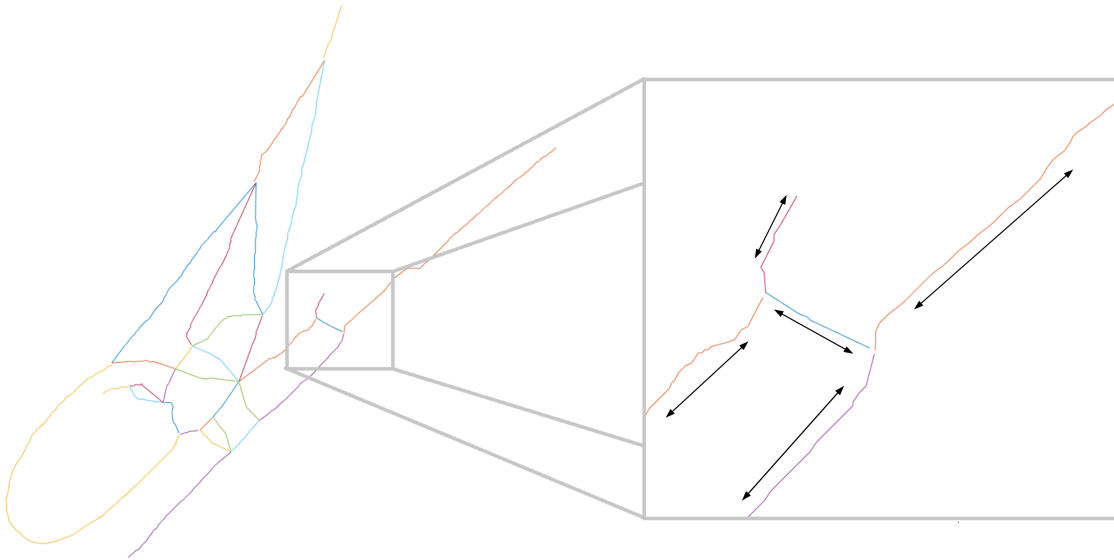


Figura 18 – Esqueleto da assinatura presente na Figura 1 com os UCSS extraídos representados pelas diferentes cores e uma aproximação ilustrando o registro dos segmentos nos dois sentidos.

3.2.2 Aplicação dos UCSS para verificação de assinaturas *offline*

Durante a elaboração desta dissertação foi desenvolvido um artigo aceito para ser publicado no ICPRAM 2017 intitulado “*Offline Signature Verification through Unambiguously Connected Skeleton Segments*” que pode ser visto no Anexo A². Nesse trabalho foram utilizadas assinaturas da base de dados MCYT-100 que, para cada usuário (total de 100 usuários), possui 25 assinaturas genuínas e 25 assinaturas forjadas. Destes grupos, 5 assinaturas genuínas, de cada usuário, são selecionadas aleatoriamente para formar o modelo do usuário. As assinaturas presentes nessa base de dados estão representadas originalmente no modo *online*, para utiliza-las no trabalho, todas elas foram convertidas em imagens e dilatadas até que as assinaturas *offline* resultantes tivessem cerca de quatro *pixels* de largura de traço. As assinaturas *offline* resultantes tiveram seu esqueleto extraído

² Esse artigo, no entanto, não foi publicado ainda, pois foi retirado do evento pelos autores, por restrições de ordem prática, e será reformulado para publicação em periódico.

utilizando um algoritmo de esqueletização clássico³ (Seção 2.1) e, em seguida, seus UCSS foram extraídos e armazenados.

Foram implementados três métodos para verificação de assinatura, todos utilizando os UCSS como característica principal das assinaturas. O primeiro foi feito para ser considerado o método base, sendo também o de raciocínio mais direto, no entanto, por ser basicamente um algoritmo de força bruta é também método de maior custo computacional. Nele cada usuário tem todas as suas assinaturas (genuínas e falsificadas, com exceção das escolhidas para o modelo) comparadas com cada assinatura do seu modelo usando DTW (*Dynamic Time Warping*) como medida de semelhança. Para isso, o modelo tem todos seus UCSS armazenados em um conjunto e, em seguida, cada UCSS da assinatura testada é associado a um UCSS do conjunto por meio da comparação entre ele e todos os UCSS do modelo usando DTW. O custo final entre uma assinatura testada e o modelo é obtido pela soma dos menores valores retornados pelo DTW para cada UCSS da assinatura testada dividido pelo número de UCSS's da mesma. Esse valor é então normalizado pela semelhança cruzada entre as assinaturas do modelo.

Com o objetivo de reduzir a quantidade de dados processados e melhorar os resultados do processamento, o segundo método foi proposto. Nele ao invés de utilizar o UCSS inteiro como característica, utiliza apenas o comprimento deles. O resultado disso é que não é necessária a utilização do DTW, que é substituído pela diferença de comprimentos como medida de custo.

Vendo uma melhoria nos resultados do segundo método em relação ao primeiro, foi criado um terceiro método. Nele os UCSS são subamostrados em quatro pontos equidistantes e são convertidos em vetores 8D, em que as dimensões representam, respectivamente, as posições em “x” e as posições em “y” dos pontos subamostrados, ou seja, $u = \{x_1, x_2, x_3, x_4, y_1, y_2, y_3, y_4\}$, sendo “u” o UCSS subamostrado. Pelo fato de todos os “u” terem as mesmas dimensões, é possível fazer a comparação direta entre eles usando distância euclidiana. Assim, o custo final de uma assinatura testada é obtido pelo mesmo processo do primeiro método substituindo o DTW pela distância euclidiana entre os “u”. Adicionalmente, foi feita uma fusão do segundo e terceiro método por meio de uma média aritmética dos seus custos finais, que foi considerado como o quarto método. Todo esse processo está devidamente descrito no anexo 3.2.2.

³ Este método de esqueletização foi escolhido pois, como será visto nas seções seguintes, o algoritmo de Zou e Yan (2001) apesar de gerar uma representação mais fiel da assinatura, possui um custo de processamento bastante alto em relação ao algoritmo clássico e não consegue eliminar totalmente os artefatos que seriam gerados usando os algoritmos de afinamento.

4 Reconstrução da Trajetória da Assinatura

Neste capítulo, inicialmente serão apresentados alguns trabalhos que trataram de recuperar a ordem de escrita de assinaturas a partir de assinaturas *offline* e influenciaram bastante no desenvolvimento deste trabalho. Ao final do capítulo será apresentado detalhadamente o método desenvolvido neste trabalho.

4.1 Trabalhos relevantes

Recuperar a trajetória que a ponta de uma caneta percorreu enquanto escrevia uma assinatura é algo que vem sendo pesquisado desde o final da década de 1980, quando Lee e Pan (1989) conseguiram traçar uma representação ordenada de desenhos arbitrários feitos por pessoas. Pelos motivos apresentados no Capítulo 1, muito trabalhos foram desenvolvidos na década de 1990, sendo o primeiro deles o trabalho de Pan e Lee (1991). Nesse trabalho (um artigo de duas páginas), foram introduzidas regras heurísticas que, segundo o autor, fazem com que o sistema siga a trajetória de modo semelhante ao que uma pessoa seguiria. No ano seguinte essas regras foram melhor explicadas em (LEE; PAN, 1992).

Apesar de não tratarem diretamente de assinaturas, Doermann e Rosenfeld (1992) e Boccignone et al. (1993) foram de grande importância para a temática de recuperação de trajetória de caneta em textos *offline*, influenciando vários trabalhos futuros, inclusive sobre assinaturas. Doermann e Rosenfeld (1992) apresentam um conjunto de pistas locais e globais de onde o *stroke*¹ se inicia e termina, regiões de junção, pistas sobre informação de velocidade e pressão, entre outros, com o objetivo de demonstrar que é factível recuperar a ordem de escrita em textos *offline* a partir de investigação detalhada dessas pistas. Já Boccignone et al. (1993) implementa um sistema de recuperação utilizando um critério de continuidade como decisão se dois segmentos do esqueleto que se encontram em um *pixel* de junção serão unidos ou não. Esse critério é definido como $G = A(\alpha) + L(\sigma) + W(\delta)$ em que, α é o ângulo entre segmentos que se ligam a um *pixel* de junção do esqueleto do texto, σ é a relação entre o segmento mais curto e o mais longo, entre os segmentos testados, e δ é o valor absoluto da diferença entre a largura média do traço dos segmentos, como pode ser visto na Figura 19. As funções $A(\alpha)$, $L(\sigma)$ e $W(\delta)$ foram obtidas empiricamente a partir de opiniões de observadores externos.

Os principais trabalhos que estimam a trajetória percorrida pela ponta da caneta em assinaturas *offline* utilizando assistência de um referencial *online* (NEL; PREEZ; HERBST,

¹ Lembrando que *stroke* no escopo deste trabalho é considerado o trecho da assinatura presente entre um evento de tocar a caneta no papel e um evento de levantar caneta do papel.

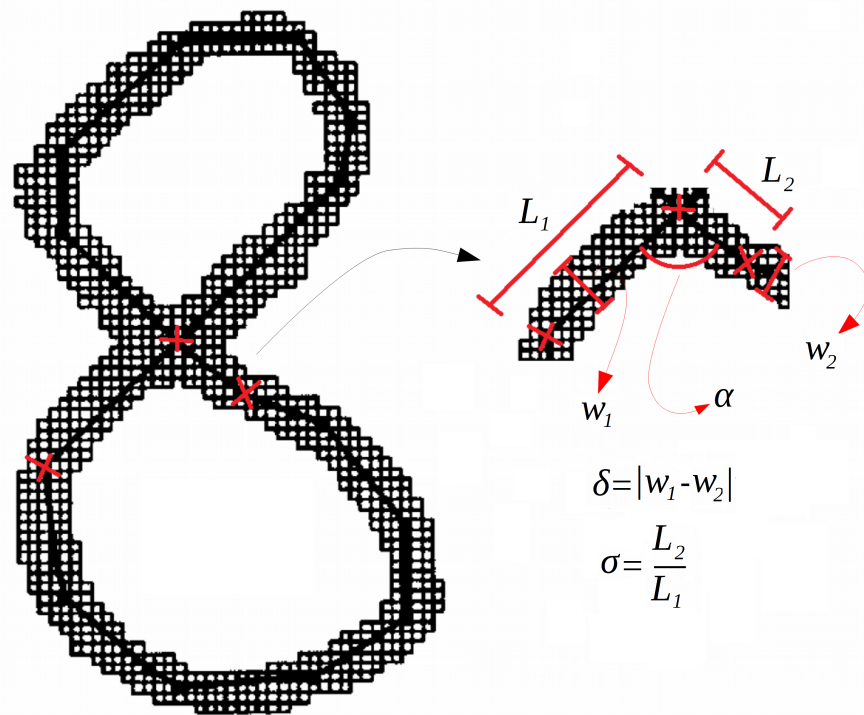


Figura 19 – Ilustração das medidas usadas para calcular a função de continuidade. Marcados com os “x” estão os pontos que ancoram os segmentos usados para essa ilustração, a região de encontro é separada à direita e suas medidas são explicitadas. Adaptado de (BOCCIGNONE et al., 1993).

2005; QIAO; LIU; TANG, 2007) baseiam-se em modelos de Markov ou alguma variação dela. Nel, Preez e Herbst (2005) utiliza um HMM (*Hidden Markov Model*) de segunda ordem com restrições de conexão para modelar a assinatura digitalizada e utiliza o algoritmo de Viterbi para estimar a trajetória da caneta na assinatura *offline*, usando os pontos do esqueleto como estados e os pontos da assinatura *online* superamostrada como observações. O processo de modelagem da assinatura gera uma matriz de transição de dimensões com escala de 4 milhões por 4 milhões, já que o esqueletos das assinaturas digitalizadas tem em média 2000 pontos. Como consequência do algoritmo de redução de ordem do HMM, proposto por Preez (1998) e usado por Nel, Preez e Herbst (2005), o número de estados do HMM de ordem reduzida cresce em quantidade proporcional à potência 2 do número de estados original. Apesar disso, os autores afirma que a matriz de transição é esparsa e que o tempo de processamento para estimar a trajetória da caneta é de cerca de 22 segundos para um esqueleto com 1600 pontos, com os recursos práticos específicos do sistema computacional implementado por eles.

Qiao, Liu e Tang (2007) usam uma técnica semelhante ao HMM para modelar a assinatura digitalizada, o CRF (*Conditional Random Field*), proposta por Lafferty, McCallum e Pereira (2001). Os parâmetros do CRF são estimados usando as assinaturas *online* como rótulos de treinamento, sendo que cada usuário tem um conjunto de parâmetros.

Com o sistema treinado o algoritmo estima a trajetória percorrida pela caneta utilizando uma adaptação do algoritmo de Viterbi para o CRF. O sistema então é usado para verificar assinaturas *offline* usando técnicas de verificação *online*.

Observando as dimensões notáveis das matrizes de transição geradas pelo método de Nel, Preez e Herbst (2005) e a necessidade de uma fase de treinamento no método de Qiao, Liu e Tang (2007)² foi pensada uma forma de recuperar a trajetória da caneta que se baseasse em um método mais leve e sem a necessidade de fase de treinamento. O DTW (*Dynamic Time Warping*) é um algoritmo computacionalmente eficiente bastante utilizado para alinhamento de sequências, mas além disso também é possível se obter o custo de alinhamento entre os sinais, o que pode ser considerado uma medida de distância, um bom material sobre DTW pode ser visto em (MÜLLER, 2007). Essas características fizeram com que o DTW fosse elemento importante deste trabalho, no sentido de simplificar grande parte da tarefa realizada pelo HMM, como ferramenta principal na tomada de decisão nas regiões do traçado das assinaturas que são ambíguas, devido às bifurcações, terminações e cruzamentos.

Visando a simplificação dos método proposto por Nel, Preez e Herbst (2005), é proposto neste trabalho um método de reconstrução da trajetória de assinaturas *offline* que, ao invés de utilizar HMM para modelar o esqueleto e inferir custos sobre as decisões do sistema, utiliza o DTW para definir os custos de decisão e modela o esqueleto usando UCSS. Este método será descrito na seção 4.2 e na seção 5.1.2 pode ser visto que, de fato o método proposto é computacionalmente mais simples que o de Nel, Preez e Herbst (2005).

4.2 Método Proposto

O método implementado percorre uma assinatura *online* enquanto compara as regiões da mesma com os UCSS da assinatura *offline* utilizando DTW a fim de encontrar os UCSS que melhor se encaixam na trajetória de referência. Esta seção será usada para descrever detalhadamente esse método, mas antes disso é necessário que alguns termos sejam esclarecidos, são eles: distância acumulada, referencial dinâmico e exemplar dinâmico.

Distância Acumulada (DA) pode ser vista como a soma das distâncias entre os pontos consecutivos de um segmento, seja ele um UCSS ou um segmento da assinatura *online*. Uma analogia semelhante é o cálculo de distância percorrida por um robô a partir da integração dos dados recebidos por um *encoder* em suas rodas. Na Figura 20 pode ser vista uma ilustração dessa distância para um UCSS e uma assinatura *online*.

Referencial Dinâmico e Exemplar Dinâmico são dois termos relacionados às assinaturas *online* presentes na base de dados utilizada neste trabalho. Referencial Dinâmico

² Dos trabalhos estudados, só esse tem uma etapa de treinamento, Nel, Preez e Herbst (2005), por exemplo, modela seu HMM baseando-se nas características do esqueleto encontrado

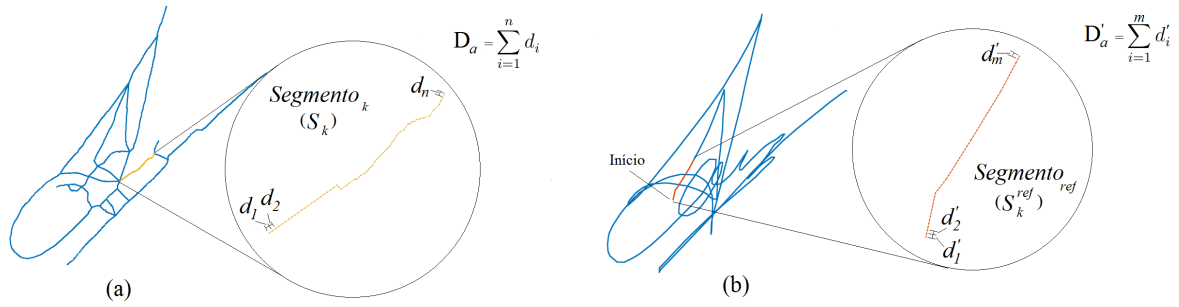


Figura 20 – Ilustração da medida de Distância Acumulada (DA) para: a) Um segmento UCSS e b) Um segmento de uma assinatura *online*.

corresponde à assinatura *online* colhida no mesmo instante da assinatura *offline*, ou seja, são a mesma assinatura colhida por dois métodos diferentes. Já um Exemplar Dinâmico é uma assinatura *online* que pertence ao mesmo autor da assinatura *offline* só que colhida em instante diferente. Na Figura 21 podem ser vistas essas assinaturas comparadas à assinatura *offline* correspondente.

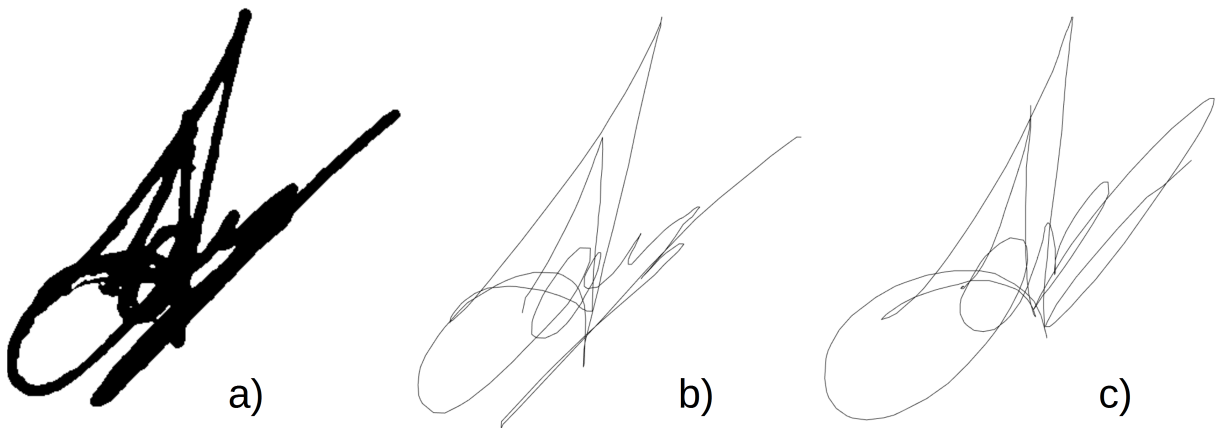


Figura 21 – a) Assinatura *offline* para ter trajetória recuperada. b) Seu referencial dinâmico. c) Um dos Exemplares Dinâmicos disponíveis na base de dados.

Antes de poder aplicar o algoritmo implementado sobre as assinaturas é feito um alinhamento entre o esqueleto da assinatura *offline* e a assinatura *online*. Para isso, o exemplar dinâmico é transladado até que sua média em “x” e “y” sejam iguais às mesmas médias do esqueleto. Após esse processo, o exemplar dinâmico é rotacionado de modo que sua orientação seja a mesma do esqueleto, isso é feito usando o processo baseado em transformada de Radon descrito em (COETZER; HERBST; PREEZ, 2004). O alinhamento é necessário pois, como o método desenvolvido é baseado em DTW, o sistema resultante, assim como o próprio DTW, é bastante sensível a esses ajustes. Na Figura 22 é possível ver uma ilustração desse processo.

Também é feita uma reamostragem dos pontos da assinatura *online*, com o objetivo

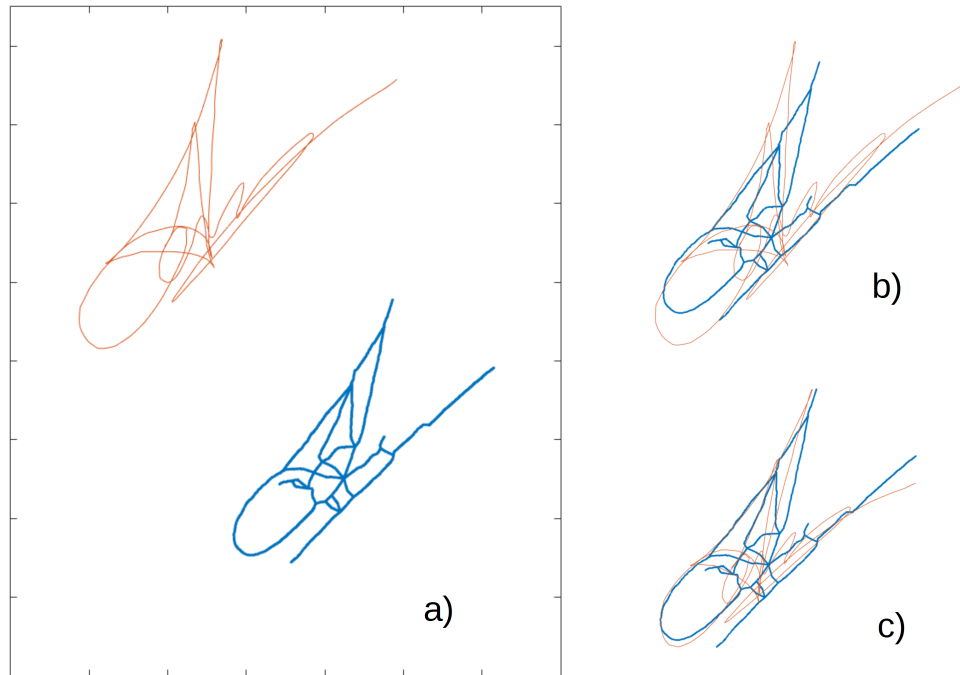


Figura 22 – a) Em laranja o Exemplar Dinâmico que será testado e em azul o esqueleto da assinatura *offline*. b) Ajuste de translação e c) Ajuste de rotação e escala.

de inserir mais pontos entre as amostras da assinatura. Isso é feito a partir de interpolações lineares entre as amostras até que não haja pontos com distância maior que uma unidade (*pixel*) entre si. A reamostragem é realizada com o objetivo de aproximar a densidade de pontos do exemplar dinâmico com a densidade de pontos do esqueleto da assinatura *offline*.

Sabendo disso, é dado início ao processo de recuperação de trajetória da assinatura *offline*. O processo é dividido em dois passos sendo o primeiro o passo de inicialização e, o segundo, o passo de reconstrução. Esses passos serão descritos nas seções 4.2.1 e 4.2.2 respectivamente.

4.2.1 Passo de inicialização

Para iniciar o processo de reconstruir a trajetória percorrida pela caneta na assinatura *offline*, primeiramente é necessário descobrir onde o movimento da caneta se inicia. O passo de inicialização é responsável por identificar essa região. Para isso, um UCSS é selecionado, este tem então sua DA (distância acumulada) calculada. A seguir, contando do primeiro ponto da assinatura *online*, é extraído um segmento que possua a DA mais próxima possível da DA do UCSS, como pode ser visto na Figura 20. Por fim, o seguimento extraído do exemplar dinâmico é comparado com o UCSS por meio de um DTW e o custo retornado é armazenado. O processo então se repete para todos os UCSS e o segmento que retornar o menor custo é considerado como região inicial do movimento.

Considerando que existem k UCSS, esse processo pode ser descrito por:

$$UCSS_0 = \underset{k}{\operatorname{argmin}}(DTW(s_k, s_k^{ref})), \quad (4.1)$$

em que s_k é o UCSS testado e s_k^{ref} é o segmento de mesma DA extraído da assinatura *online*. Consideramos o UCSS selecionado como início do *stroke* recuperado.

4.2.2 Passo de reconstrução

Encontrado o início *stroke* é dado início ao passo de Reconstrução. Nesse passo a trajetória que a caneta percorreu pelo restante do *stroke* será recuperada. Para isso, ao final do primeiro UCSS é feita uma verificação dos pontos mais próximos que não pertencem ao mesmo UCSS. Feito isso, são identificados os UCSS que contem como primeiro elemento os pontos selecionados. Para cada um desses UCSS será feito um teste que consiste em:

- 1 Concatenar o UCSS testado com o UCSS selecionado no passo de inicialização;
- 2 Computar a DA do vetor resultante;
- 3 Extrair da assinatura *online* um segmento com DA mais próximo possível da encontrada no item anterior.

Ao finalizar esse processo o UCSS que melhor se encaixa é encontrado utilizando a equação 4.1. Este UCSS é concatenado, agora definitivamente, com o UCSS inicial e o resultado disso é considerado como o primeiro UCSS (resultado obtido na seção 4.2.1).

Esse procedimento é, então, repetido até que a DA do vetor resultante das concatenações ultrapasse a DA máxima do *stroke* da assinatura *online* testada. Ao final deste processo é obtido um vetor que representa a trajetória recuperada de um dos *strokes* da assinatura *offline*. Este processo é feito para todos os presentes na assinatura *online* testada, podendo acontecer desta assinatura possuir mais *strokes* do que o Referencial Dinâmico, o que acarretará numa estranha trajetória recuperada. Por isso o uso de mais de um Exemplar Dinâmico é essencial. Na Figura 23 é possível ver o esquemático do sistema completo.

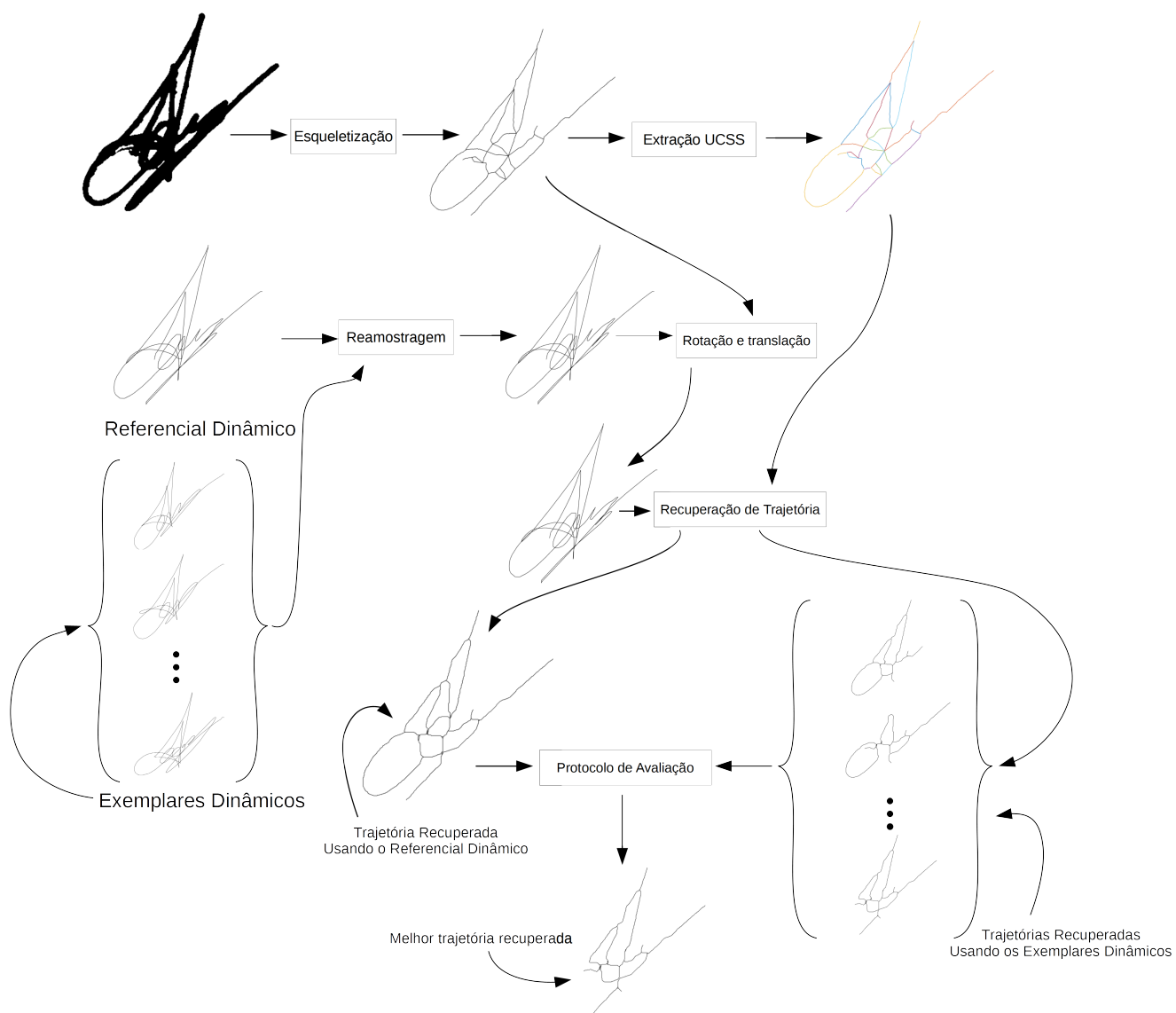


Figura 23 – Esquemático do funcionamento do sistema completo.

5 Resultados e Discussões

Neste trabalho foi utilizada a base de dados US_SIGBASE colhida por Nel (2005) e disponível em www.ussigbase.org. Esta base de dados possui a representação tanto *online* quanto *offline* de 51 assinaturas de usuários distintos, além disso outros 14 exemplares *online* foram extraídos de cada usuário para formar uma base de treinamento para algoritmos de recuperação de trajetória. Em seu trabalho, Nel (2005) utilizou os 14 exemplares *online* de cada usuário para recuperar a trajetória da assinatura *offline* deste mesmo usuário. A representação *online* da assinatura *offline* (que foi colhida no mesmo instante, logo é a mesma assinatura representada em outro meio) foi utilizada para servir de referência para os resultados do método de recuperação. O experimento implementado neste trabalho foi o mesmo aplicado em (NEL, 2005) com a diferença que o método utilizado para recuperar as trajetórias das assinaturas *offline* foi o descrito no Capítulo 4.2. Na seção 5.1 este experimento será descrito e em seguida na seção 5.1.2 serão apresentados os resultados.

Também foi desenvolvido, durante a produção desta dissertação, um trabalho para ser publicado no ICPRAM 2017 que foi aceito para publicação, no entanto o processo de publicação não foi levado a diante por restrições de ordem prática. Apesar disso, esse trabalho pode ser visto na íntegra no anexo A. Neste capítulo também serão descritos os experimentos realizados (seção 5.2) e os resultados na obtidos no artigo (seção 5.2.1).

5.1 Experimentos de Reconstrução de Trajetória

Como foi visto na seção 4.1 o trabalho de Nel, Preez e Herbst (2005) foi a principal referência para o desenvolvimento desta dissertação. Sabendo disso, fica claro que o método aqui desenvolvido deve ser comparado ao método de Nel, Preez e Herbst (2005). Para isso, foi reproduzido o mesmo experimento utilizado em (NEL; PREEZ; HERBST, 2005). Este experimento, para cada usuário, consiste em:

- 1 Aplicar o algoritmo de recuperação de trajetória da assinatura *offline* usando o Referencial e os Exemplares Dinâmicos como guia e guardar as trajetórias resultantes;
- 2 Aplicar o protocolo de avaliação da trajetória recuperada obtida no item 1 usando o resultado do Referencial Dinâmico como referência e os resultados dos Exemplares Dinâmicos como testes;
- 3 Selecionar o melhor resultado do protocolo de avaliação como medida de eficácia do sistema para cada usuário;

Tabela 1 – Comparação entre os resultados de similaridade média do trabalho de [Nel, Preez e Herbst \(2005\)](#) com o sistema proposto nesta dissertação usando: o método de esqueletização descrito na seção 2.3 e o método de esqueletização clássica.

Método de recuperação de trajetória	ZYM	Esqueletização Clássica
Nel, Preez e Herbst (2005)	91,0%	90,3%
Método proposto	70,4%	71,8%

- 4 Calcular a média dos melhores resultados como desempenho para o sistema completo.

Como pode ser visto, o experimento é bastante direto, com exceção do protocolo de avaliação que será descrito na seção 5.1.1.

5.1.1 Protocolo de Avaliação da Reconstrução

O resultado do item 1 do experimento descrito na seção 5.1 frequentemente possui quantidades de pontos diferentes, para lidar adequadamente com isso, os vetores numéricos que representam as assinaturas foram alinhados entre si utilizando um DTW. Estes vetores são, então, comparados diretamente ponto a ponto (distância euclidiana) o que gera um vetor de erros $e(t)$. Esse vetor possui uma característica bastante ruidosa proveniente das pequenas diferenças entre as assinaturas recuperadas (característica da escrita humana), no entanto, existem regiões desse vetor com magnitude muito maior que o restante dele. Essas regiões, segundo [Nel, Preez e Herbst \(2005\)](#), são os pontos em que houve erro de reconstrução. Para identificar essas regiões efetivamente, é aplicado um filtro de abertura e fechamento morfológico ([GONZALEZ; WOODS, 2007](#)) com janela $w = 10$ sobre o vetor $e(t)$, o resultado disso é um vetor $e_f(t)$ com a demarcação de onde o algoritmo de reconstrução errou. Esse vetor também pode ser usado como medida de similaridade, para isso, é computada a quantidade de pontos de $e_f(t)$ que são iguais a zero e este valor é dividido pela quantidade de pontos em $e_f(t)$ obtendo uma porcentagem de similaridade. Na Figura 24 pode ser vista uma ilustração de todo esse processo.

5.1.2 Resultados do experimento de reconstrução

O experimento descrito nas seções anteriores foi aplicado utilizando dois métodos de esqueletização, a esqueletização clássica apresentada na seção 2.1 e o método de esqueletização apresentado na seção 2.3 (ZYM). Estes resultados são então comparados com os resultados obtidos por [Nel, Preez e Herbst \(2005\)](#) na Tabela 1.

Como pode ser visto, baseado no protocolo de avaliação de ([NEL; PREEZ; HERBST, 2005](#)) o sistema proposto nesta dissertação possui um desempenho muito abaixo do encontrado no trabalho original. No entanto, apesar dessa diferença, o sistema aqui proposto é significativamente mais simples sendo que, para o sistema de ([NEL; PREEZ;](#)

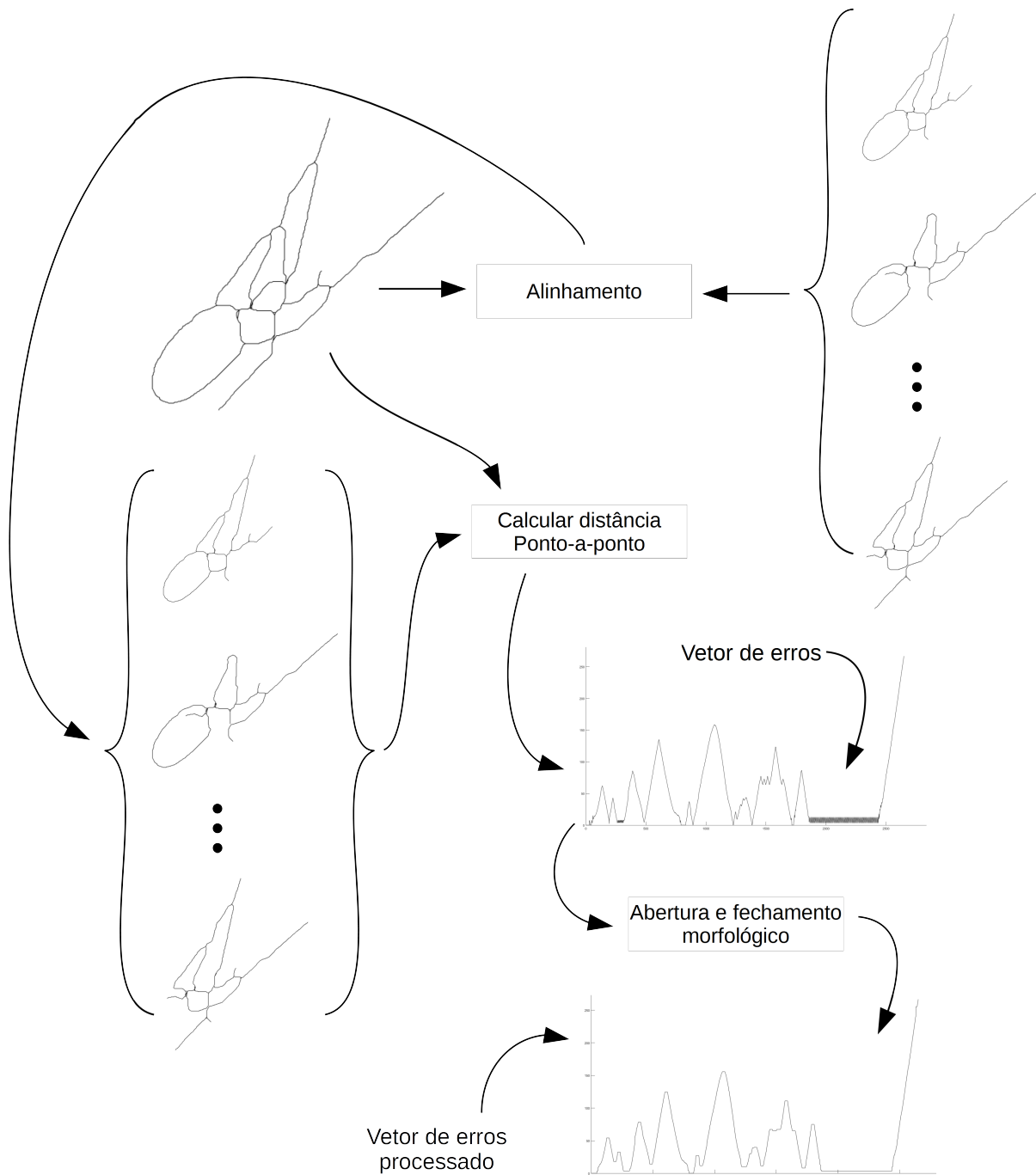


Figura 24 – Ilustração do protocolo de avaliação.

[HERBST, 2005](#)), para um esqueleto de 1600 pontos, é necessário aplicar um algoritmo de Viterbi em uma matriz de transição com dimensão de 2560000×2560000 . No método proposto, a única operação necessária é um DTW aplicado a cada iteração em que a maior dimensão de vetor encontrada durante todos os experimentos foi 9000×2 .

Uma característica do protocolo de avaliação de Nel, é que nele não é medido diretamente o quão bem foi feita a recuperação da trajetória da assinatura *offline*. O que é medido é quão parecida a assinatura recuperada é da trajetória recuperada, usando o

referencial dinâmico. Assim, a medida de desempenho explicada em 5.1.1 faz com que trajetórias esdrúxulas sejam tomadas como corretas como ilustrado na Figura 25. Sabendo disso, foram notados alguns defeitos no método aqui proposto que podem ser listados como:

- 1 Traços interrompidos ou quebrados na assinatura *offline* causam erro de recuperação de trajetória (Figura 25);
- 2 O sistema é muito sensível ao alinhamento da assinatura *online* com o esqueleto;
- 3 Foi verificada a necessidade de mais informações no momento de decisão de qual direção seguir, quando, em alguns casos, o sistema ficava preso em uma região por defeito de decisão;

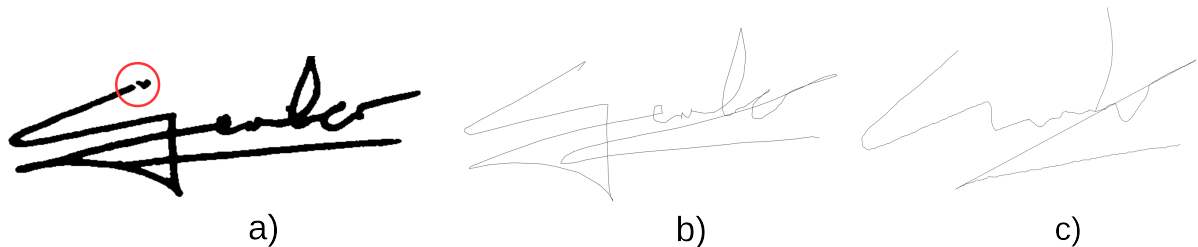


Figura 25 – a) Assinatura *offline* com falha no traço marcado pelo círculo. b) Referencial Dinâmico de (a).c) Trajetória recuperada erroneamente devido ao corte no traço.

No trabalho de Nel, Preez e Herbst (2005) também foram divulgados dados sobre o tempo de processamento. Para um esqueleto de 1600 pontos seu sistema demora 16 segundos para modelar a assinatura *offline* e 22 segundos para estimar a trajetória percorrida baseando-se em um exemplar dinâmico. O sistema desenvolvido aqui demora, em média, 11 segundos para modelar a assinatura *offline* e 5 segundos para estimar a trajetória. Não é possível, comparar diretamente as durações acima relatadas (servindo portanto como medidas aproximadas dos custos de cada método) ao de Nel, Preez e Herbst (2005) em termos de tempo de processamento devido à diferença de *hardware* computacional decorrente dos mais de 10 anos entre os trabalhos, além disso, o sistema proposto nessa dissertação foi implementado em MATLAB, que é uma linguagem de programação interpretada, o que a torna frequentemente mais lenta do que outras linguagens de mais baixo nível.

A obtenção de melhores resultados utilizando a esqueletização clássica comparada à esqueletização mais sofisticada foi bastante surpreendente. No entanto, a diferença de desempenho não foi muito grande, característica que pode ser vista também no trabalho de Nel, Preez e Herbst (2005), no sentido inverso. Isso sugere que, nesta base de dados, os

dois métodos de esqueletização geram esqueletos bastante semelhantes, além disso, por causa desses resultados, o algoritmo clássico de esqueletização se torna mais atraente à utilização por causa de sua baixa complexidade computacional.

5.2 Experimentos de Verificação Biométrica

No artigo desenvolvido durante a dissertação, onde se visou a verificação biométrica de identidade de indivíduos através de suas assinaturas offline, a base de dados utilizada foi, no entanto, e por razões que serão explicadas mais adiante, a MCYT-100 (disponível em <http://atvs.ii.uam.es/mcyt100f.html>), que é uma base de assinaturas *online* bastante utilizada em biometria, contendo 50 versões de assinaturas de 100 usuários diferentes sendo que, para cada usuário, 25 assinaturas são genuínas e 25 são forjadas. Utilizando esta base, os métodos descritos na seção 3.2.2 foram testados com o objetivo de verificar qual deles conseguia melhor determinar a autenticidade ou não de uma assinatura *offline*. A existência de vasto referencial relativo ao desempenho de métodos de verificação baseados em assinaturas *online* serve como base para estabelecer comparações (com a mesma base) entre taxas de verificações para assinaturas *online* e *offline*. É aceito que a verificação com assinaturas *online* leva a taxas muito melhores, por volta de 95% de verificações corretas, para a base MCYT-100. Então espera-se que essa taxa possa servir como um teto, ou uma meta para os métodos baseados em assinaturas *offline*, sobre versões *offline* de assinaturas da mesma base.

Por essa razão, isto é, para permitir que os resultados publicados sobre a MCYT-100 sirvam como meta para este trabalho, quando possível, foram seguidos os mesmos protocolos típicos dos experimentos de verificação de assinaturas *online*, a saber:

- 1 Para cada usuário foram selecionadas, aleatoriamente, 5 assinaturas genuínas para formar um modelo da assinatura do usuário;
- 2 As 45 assinaturas restantes (do mesmo usuário) foram testadas utilizando os métodos descritos na seção 3.2.2, retornando medidas numéricas associadas;
- 3 As medidas encontradas no passo 2 foram limiarizadas utilizando um limiar variável para definir se as assinaturas são genuínas ou forjadas;
- 4 Conhecendo a natureza das assinaturas testadas (se são genuínas ou forjadas), são estimadas as taxas de Falsa Aceitação (FAR) e Falsa Rejeição (FRR);
- 5 Os passos de 1 a 4 foram repetidos para todos os usuários a fim de obter um resultado geral sobre as taxas FAR e FRR;

- 6 Esse processo foi repetido 50 vezes (cada repetição possuiu um conjunto diferente de assinaturas genuínas no modelo) a fim de reduzir enviesamento decorrente da escolha das assinaturas do modelo;
- 7 Os passos de 1 a 6 foram repetidos para cada um dos métodos descritos na seção 3.2.2.

Com esse processo é possível fazer uma comparação entre a eficiência de cada método proposto no artigo. Para obter mais uma referência de comparação, foram elaborados cartões que foram usados para colher a opinião de pessoas quanto à veracidade ou não das assinaturas. Esses cartões continham 5 assinaturas genuínas de um lado da folha, para servir de referência para o voluntário. Em seguida, era solicitado ao voluntário que verificasse quais das 10 assinaturas (5 genuínas e 5 forjadas) do outro lado da folha eram genuínas ou forjadas. Ao voluntário, não era informado o número de assinaturas verdadeiras no cartão, embora esse número fosse sempre 5, para todos os cartões gerados. Para cada usuário da base de dados foi criado um cartão, estes foram fotocopiados e distribuídos para voluntários (compostos por alunos e professores da universidade) devidamente instruídos sobre como preencher os cartões. É possível ver um exemplo destes cartões na Figura 26. Os resultados dessa coleta foram computados para obter a Pontuação Média de Opinião que será denotada como MOS (*Mean Opinion Scores*).

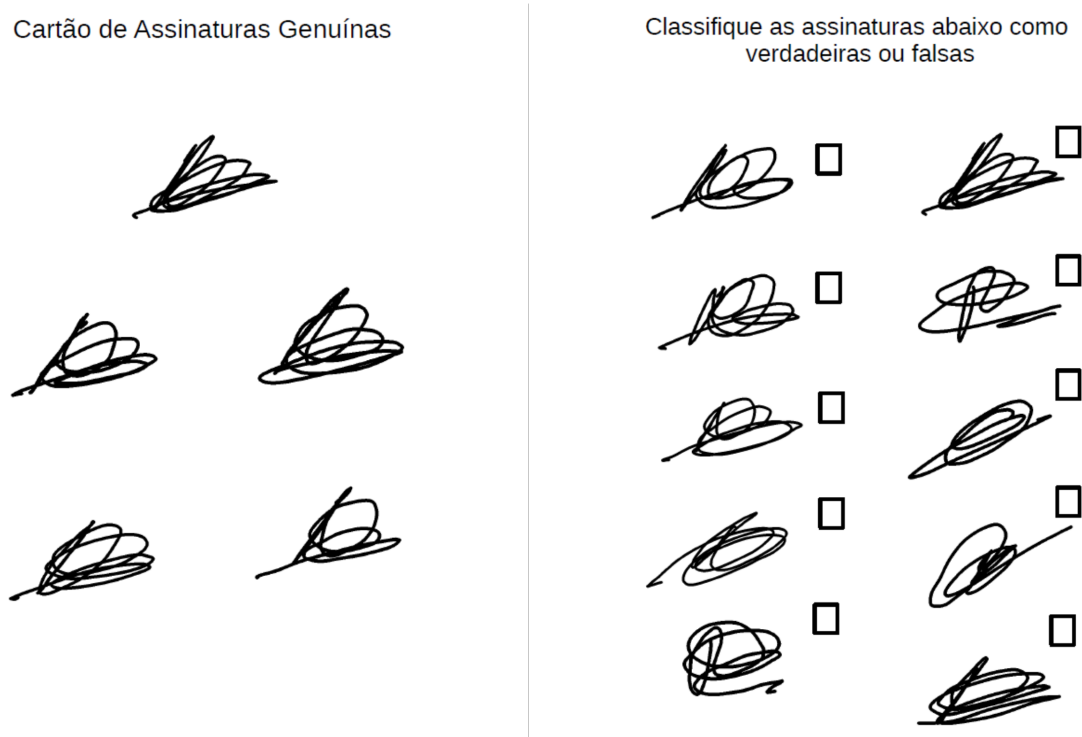


Figura 26 – Exemplo de um dos cartões distribuídos para os voluntários utilizados para obter a MOS

Tabela 2 – Resultado de cada método após 50 execuções independentes. Cada execução corresponde a um grupo aleatório de 5 assinaturas genuínas selecionadas para o modelo de cada usuário.

Método	EER	Desvio Padrão
I-Método base	32,2%	1,4%
II-Baseado em Comprimento de UCSS	25,1%	1,2%
III-Baseado em UCSS subamostrado	19,4%	1,3%
IV-Fusão (média) dos custos de II e III	18,7%	1,0 %

5.2.1 Resultados do Experimento de Verificação Biométrica

Após alguns dias de coleta de opinião usando os cartões descritos na seção anterior, foram totalizados 239 cartões preenchidos por 103 voluntários. O processamento desses cartões gerou uma taxa de falsa aceitação (FAR) de 11,0% e uma taxa de falsa rejeição (FRR) de 25,6%, esses valores foram considerados como MOS. Os experimentos descritos na seção 5.2 foram expressos de duas maneiras. Primeiro as taxas FAR e FRR obtidas no passo 5 descrito na seção 5.2 foram expressos em um gráfico (visível na Figura 27) para melhor visualizar a diferença de desempenho entres os métodos. E por último, foram encontradas as Taxas de Erro Igual chamadas EER (*Equal Error Rate*) utilizando os dados resultantes do passo 7 da seção 5.2, esses resultados podem ser vistos na Tabela 2.

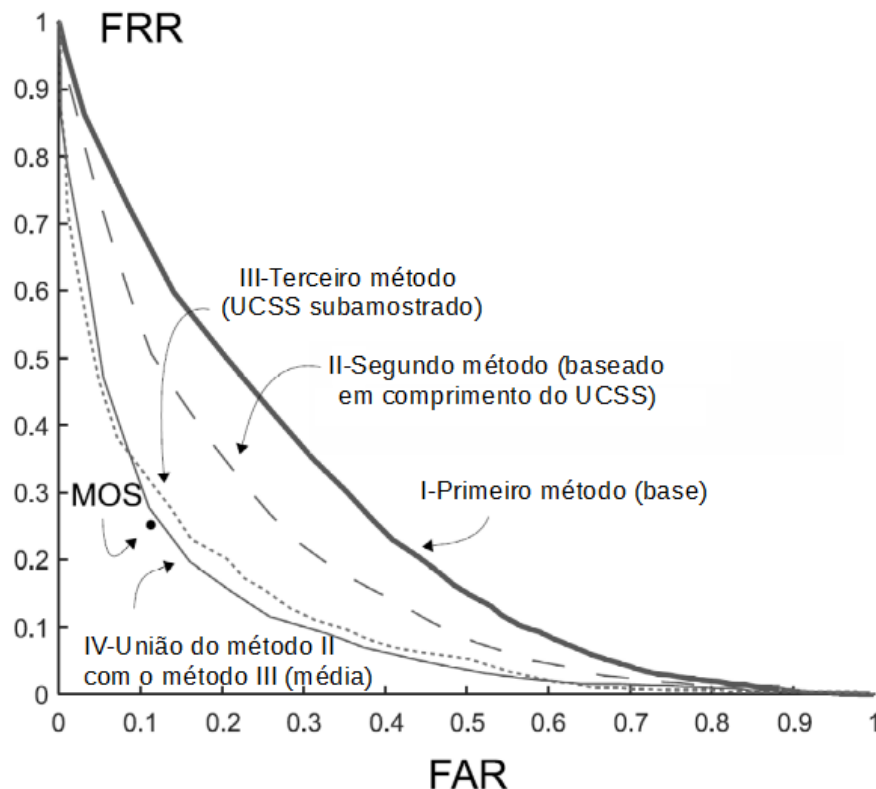


Figura 27 – Exemplo de um dos cartões distribuídos para os voluntários utilizados para obter a MOS

Pode ser visto, pela Figura 27 e pela tabela 2, que mesmo assumindo que o UCSS completo possui mais informações sobre o formato da assinatura, a comparação direta entre eles obteve um resultado pior do que quando o UCSS foi subamostrado em apenas 4 pontos. Mais surpreendente ainda foi o fato de usar apenas o comprimento dos UCSS já ter melhorado bastante o desempenho do sistema. Esses resultados são pistas bastante fortes de que a informação de formato do UCSS não é relevante para a tarefa de verificação biométrica. Essa informação de formato pode, inclusive ser considerada como um tipo de ruído para o sistema biométrico havendo também a possibilidade de que esse excesso de informação esteja causando alguns dos problemas encontrados no sistema desenvolvido na dissertação.

6 Conclusões

Neste trabalho foi inicialmente feito um estudo geral sobre o tema de recuperação de trajetória em textos e assinaturas *offline*. Nesse estudo, foram vistas diversas técnicas e passos necessários para realizar esta tarefa, incluindo métodos que utilizavam assinaturas *online* previamente para auxílio ao algoritmo de recuperação de trajetória.

Através desse estudo, foi selecionada uma metodologia de trabalho baseada em dividir o processo de recuperar a trajetória da assinatura em três passos, sendo eles: (i) computar o esqueleto da assinatura, (ii) extrair as características necessárias para então, (iii) recuperar a trajetória. A fim de compreender melhor cada um desses passos, foi feita uma revisão bibliográfica específica para cada um deles observando os principais trabalhos e implementando os que fossem interessantes para o desenvolvimento desta dissertação. Com isso, foi possível encontrar problemas em aberto para cada um desses passos como, por exemplo, encontrar um método ideal para extrair o esqueleto de uma assinatura, como utilizar uma assinatura *online* para estimar a trajetória percorrida pela caneta em uma assinatura *offline* de maneira rápida e eficiente? Ou então qual a melhor maneira de extrair características do esqueleto de uma assinatura? Neste último problema em aberto foi possível gerar alguma contribuição útil ao formalizar o método de extração baseado em UCSS.

Conhecendo a metodologia, os algoritmos e métodos considerados aplicáveis ao trabalho foram desenvolvidos e, com um sistema idealizado e implementado computacionalmente, foram elaborados dois experimentos: um que permitiu a comparação de resultados com um trabalho referência desta dissertação (NEL; PREEZ; HERBST, 2005). E um segundo experimento relacionado diretamente ao conceito de UCSS, aplicado diretamente à verificação biométrica (que é, de fato, o objetivo final de muitos métodos de recuperação de trajetória de assinaturas *offline*). Além da usual comparação de desempenho entre métodos, o resultado dessa verificação foi comparado também ao desempenho humano ao realizar a mesma tarefa.

No primeiro experimento, foram obtidos resultados relativamente ruins para o método proposto, sob o critério usado no trabalho referenciado, para a mesma base de assinaturas. No entanto, foi ilustrado na seção 5.1.2 que protocolo de avaliação do trabalho referenciado mostrou-se problemático na comparação de trajetórias claramente diferentes, recuperadas de uma referência e de uma assinatura de teste. Como trabalho futuro, acreditamos que seja necessária a substituição desse critério por outro que compare diretamente a trajetória original com os resultados do algoritmo de recuperação. Também foi notado que há pouca diferença entre os resultados utilizando os dois métodos de

esqueletização, sendo que, neste trabalho, o método clássico obteve média de resultados melhores que o método mais sofisticado. Usando o método de [Nel, Preez e Herbst \(2005\)](#) e os dois métodos de esqueletização, também foram obtidos resultados semelhantes, no entanto, o melhor resultado foi encontrado usando o algoritmo de esqueletização mais sofisticado. A diferença de desempenho sutil, comparada à grande diferença de complexidade computacional, sugere que, pela sua menor complexidade computacional, é melhor utilizar o método clássico.

Apesar de não obter resultados de recuperação de trajetória satisfatórios, o algoritmo aqui implementado se mostrou bem menos complexo que o desenvolvido no trabalho de referência, isso gera motivação para continuar explorando essa abordagem porém, futuramente, incluindo mais informações no momento de tomada de decisão, além de usar apenas o DTW.

A parte do trabalho voltada à verificação biométrica, por outro lado, se mostrou promissora ao utilizar uma forma diferente de extrair características das assinaturas *offline* usando os UCSS, além disso, também foi levantada a hipótese de que a informação completa de formato do UCSS pode não ser relevante para propósitos biométricos. Mais estudo sobre essa abordagem são necessários ainda para que haja uma melhor compreensão acerca da relevância do contorno detalhado das UCSS na recuperação de trajetórias e aplicações além da verificação biométrica, onde começou-se a perceber que o comprimento e o formato aproximado das UCSS parecem carregar a maior parte da informação relevante sobre a identidade daquele que assina.

Referências

- BAATI, A. E. et al. Recovery of temporal information from off-line arabic handwritten. In: IEEE. *The 3rd ACS/IEEE International Conference on Computer Systems and Applications, 2005*. [S.l.], 2005. p. 127. Citado 3 vezes nas páginas [V](#), [20](#) e [21](#).
- BOCCIGNONE, G. et al. Recovering dynamic information from static handwriting. *Pattern Recognition*, Elsevier, v. 26, n. 3, p. 409–418, 1993. Citado 8 vezes nas páginas [IV](#), [V](#), [3](#), [8](#), [9](#), [10](#), [24](#) e [25](#).
- CANUTO, J. C. *Biomechanical online signature modeling applied to verification*. Tese (Doutorado) — Institut National des Télécommunications, 2014. Citado na página [1](#).
- CHANG, H.-H.; YAN, H. Analysis of stroke structures of handwritten chinese characters. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, IEEE, v. 29, n. 1, p. 47–61, 1999. Citado na página [3](#).
- CHEW, L. P. Constrained delaunay triangulations. *Algorithmica*, Springer, v. 4, n. 1-4, p. 97–108, 1989. Citado na página [13](#).
- COETZER, J.; HERBST, B. M.; PREEZ, J. A. du. Offline signature verification using the discrete radon transform and a hidden markov model. *EURASIP Journal on Advances in Signal Processing*, Springer, v. 2004, n. 4, p. 1–13, 2004. Citado na página [27](#).
- DAWOUD, A.; KAMEL, M. New approach for the skeletonization of handwritten characters in gray-level images. In: IEEE COMPUTER SOCIETY. *Proceedings of the Seventh International Conference on Document Analysis and Recognition-Volume 2*. [S.l.], 2003. p. 1233. Citado 3 vezes nas páginas [IV](#), [11](#) e [12](#).
- DOERMANN, D. S.; ROSENFELD, A. Recovery of temporal information from static images of handwriting. In: *Proceedings 1992 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. [S.l.: s.n.], 1992. p. 162–168. ISSN 1063-6919. Citado na página [24](#).
- GONZALEZ, R. C.; WOODS, R. E. Image processing. *Digital image processing*, v. 2, 2007. Citado 5 vezes nas páginas [IV](#), [7](#), [8](#), [10](#) e [32](#).
- GUO, J. K.; DOERMANN, D.; ROSENFELD, A. Forgery detection by local correspondence. *International Journal of Pattern Recognition and Artificial Intelligence*, World Scientific, v. 15, n. 04, p. 579–641, 2001. Citado 2 vezes nas páginas [4](#) e [5](#).
- HUANG, T.; YASUHARA, M. A total stroke slalom method for searching for the optimal drawing order of off-line handwriting. In: IEEE. *Systems, Man and Cybernetics, 1995. Intelligent Systems for the 21st Century., IEEE International Conference on*. [S.l.], 1995. v. 3, p. 2789–2794. Citado 3 vezes nas páginas [4](#), [8](#) e [17](#).
- HUANG, W.; RONG, G.; BIAN, Z. Strokes recovering from static handwriting. In: IEEE. *Document Analysis and Recognition, 1995., Proceedings of the Third International Conference on*. [S.l.], 1995. v. 2, p. 861–864. Citado 4 vezes nas páginas [V](#), [3](#), [17](#) e [18](#).

JÄGER, S. Recovering dynamic information from static, handwritten word images. Citeseer, 1998. Citado na página 4.

KATO, Y.; YASUHARA, M. Recovery of drawing order from single-stroke handwriting images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, IEEE, v. 22, n. 9, p. 938–949, 2000. Citado 2 vezes nas páginas 3 e 4.

KÉGL, B.; KRZYZAK, A. Piecewise linear skeletonization using principal curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, IEEE, v. 24, n. 1, p. 59–74, 2002. Citado 3 vezes nas páginas IV, 10 e 11.

LAFFERTY, J.; MCCALLUM, A.; PEREIRA, F. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: *Proceedings of the eighteenth international conference on machine learning, ICML*. [S.l.: s.n.], 2001. v. 1, p. 282–289. Citado na página 25.

LALLICAN, P. M.; VIARD-GAUDIN, C. A kalman approach for stroke order recovering from off-line handwriting. In: IEEE. *Document Analysis and Recognition, 1997., Proceedings of the Fourth International Conference on*. [S.l.], 1997. v. 2, p. 519–522. Citado na página 3.

LALLICAN, P. M.; VIARD-GAUDIN, C.; KNERR, S. From off-line to on-line handwriting recognition. In: *Proceedings of the Seventh International Workshop on Frontiers in Handwriting Recognition*. [S.l.: s.n.], 2000. p. 303–312. Citado na página 4.

LAM, L.; LEE, S.-W.; SUEN, C. Y. Thinning methodologies-a comprehensive survey. *IEEE Transactions on pattern analysis and machine intelligence*, IEEE Computer Society, v. 14, n. 9, p. 869–885, 1992. Citado na página 9.

LAU, K. K.; YUEN, P. C.; TANG, Y. Y. Recovery of writing sequence of static images of handwriting using uwm. In: *ICDAR*. [S.l.: s.n.], 2003. p. 1123–1127. Citado 2 vezes nas páginas 4 e 5.

LEE, S.; PAN, J. Tracing and representation of human line drawings. In: IEEE. *Systems, Man and Cybernetics, 1989. Conference Proceedings., IEEE International Conference on*. [S.l.], 1989. p. 1055–1061. Citado na página 24.

LEE, S.; PAN, J. C. Offline tracing and representation of signatures. *Systems, Man and Cybernetics, IEEE Transactions on*, IEEE, v. 22, n. 4, p. 755–771, 1992. Citado 6 vezes nas páginas 3, 8, 9, 17, 18 e 24.

LI, C. et al. On-line signature verification method by laplacian spectral analysis and dynamic time warping. In: INTERNATIONAL SOCIETY FOR OPTICS AND PHOTONICS. *IS&T/SPIE Electronic Imaging*. [S.l.], 2013. p. 90210S–90210S. Citado na página 1.

MÜLLER, M. Dynamic time warping. *Information retrieval for music and motion*, Springer, p. 69–84, 2007. Citado na página 26.

MUNICH, M. E.; PERONA, P. Visual identification by signature tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, IEEE, v. 25, n. 2, p. 200–217, 2003. Citado na página 2.

- NEL, E.; PREEZ, J. A. du; HERBST, B. M. Estimating the pen trajectories of multi-path static scripts using hidden markov models. In: IEEE. *Document Analysis and Recognition, 2005. Proceedings. Eighth International Conference on*. [S.l.], 2005. p. 41–45. Citado 19 vezes nas páginas [V](#), [VII](#), [2](#), [3](#), [4](#), [7](#), [8](#), [12](#), [17](#), [18](#), [19](#), [25](#), [26](#), [31](#), [32](#), [33](#), [34](#), [39](#) e [40](#).
- NEL, E.-M. *Estimating the pen trajectories of static handwritten scripts using hidden Markov models*. Tese (Doutorado) — Stellenbosch: University of Stellenbosch, 2005. Citado 6 vezes nas páginas [IV](#), [1](#), [3](#), [4](#), [15](#) e [31](#).
- NGUYEN, V.; BLUMENSTEIN, M. Techniques for static handwriting trajectory recovery: a survey. In: ACM. *Proceedings of the 9th IAPR International Workshop on Document Analysis Systems*. [S.l.], 2010. p. 463–470. Citado 4 vezes nas páginas [IV](#), [1](#), [5](#) e [6](#).
- PAN, J. C.; LEE, S. Offline tracing and representation of signatures. In: IEEE. *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on*. [S.l.], 1991. p. 679–680. Citado na página [24](#).
- PERVOUCHINE, V.; LEEDHAM, G.; MELIKHOV, K. Handwritten character skeletonisation for forensic document analysis. In: ACM. *Proceedings of the 2005 ACM symposium on Applied computing*. [S.l.], 2005. p. 754–758. Citado 2 vezes nas páginas [3](#) e [11](#).
- PLAMONDON, R.; PRIVITERA, C. M. The segmentation of cursive handwriting: an approach based on off-line recovery of the motor-temporal information. *IEEE Transactions on Image Processing*, IEEE, v. 8, n. 1, p. 80–91, 1999. Citado 3 vezes nas páginas [V](#), [19](#) e [20](#).
- PLAMONDON, R.; SRIHARI, S. N. Online and off-line handwriting recognition: a comprehensive survey. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, IEEE, v. 22, n. 1, p. 63–84, 2000. Citado 3 vezes nas páginas [1](#), [2](#) e [4](#).
- PREEZ, J. A. du. Efficient training of high-order hidden markov models using first-order representations. *Computer speech & language*, Elsevier, v. 12, n. 1, p. 23–39, 1998. Citado na página [25](#).
- QIAO, Y.; LIU, J.; TANG, X. Offline signature verification using online handwriting registration. In: IEEE. *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*. [S.l.], 2007. p. 1–8. Citado 7 vezes nas páginas [1](#), [2](#), [4](#), [8](#), [17](#), [25](#) e [26](#).
- QIAO, Y.; NISHIARA, M.; YASUHARA, M. A framework toward restoration of writing order from single-stroked handwriting image. *IEEE transactions on pattern analysis and machine intelligence*, IEEE, v. 28, n. 11, p. 1724–1737, 2006. Citado 4 vezes nas páginas [V](#), [17](#), [18](#) e [19](#).
- SAHA, P. K.; BORGEFORS, G.; BAJA, G. S. di. A survey on skeletonization algorithms and their applications. *Pattern Recognition Letters*, Elsevier, v. 76, p. 3–12, 2016. Citado na página [7](#).
- SAIKIA, H.; SARMA, K. C. Approaches and issues in offline signature verification system. *International Journal of Computer Applications*, Citeseer, v. 42, n. 16, p. 45–52, 2012. Citado na página [1](#).

- SCHAFER, B.; VIRIRI, S. An off-line signature verification system. In: IEEE. *Signal and Image Processing Applications (ICSIPA), 2009 IEEE International Conference on*. [S.l.], 2009. p. 95–100. Citado na página 1.
- STEINHERZ, T.; INTRATOR, N.; RIVLIN, E. A special skeletonization algorithm for cursive words. In: *Proc. Int'l Workshop Frontiers in Handwriting Recognition*. [S.l.: s.n.], 2000. p. 529–534. Citado 2 vezes nas páginas 10 e 11.
- WIROTIUS, M.; SEROPIAN, A.; VINCENT, N. Writer identification from gray level distribution. In: *ICDAR*. [S.l.: s.n.], 2003. p. 1168–1172. Citado na página 3.
- ZOU, J. J.; YAN, H. Skeletonization of ribbon-like shapes based on regularity and singularity analyses. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, IEEE, v. 31, n. 3, p. 401–407, 2001. Citado 7 vezes nas páginas 7, 8, 12, 13, 14, 15 e 23.

Anexos

ANEXO A – Artigo produzido pelo autor

Offline Signature Verification through Unambiguously Connected Skeleton Segments

Keywords: Offline signatures, mean opinion score, signature segmentation, biometrics.

Abstract: A method for offline signature verification is presented in this paper. It is based on the segmentation of the signature image skeleton into unambiguous sequences of points. These segments are compactly encoded as sets of 9 scalars (8 for shape and position, and 1 for length representation). Signature authenticity is inferred through Euclidean distance comparisons between pairs of segments. The average performance of the proposed method is evaluated through experiments with offline versions of signatures from the MCYT-100 database. For comparison purposes, two reference approaches are applied to the same set of signatures, one based on Dynamic Time Warping distances between segments, and another based on average human performance under equivalent experimental protocol. Results suggest that, by discarding segment shape details (as embodied by the proposed encoding), verification performance is improved.

1 Introduction

The handwritten signature is a form of personal identification widely accepted, both socially and legally, and it has been used for centuries to authenticate documents such as bank checks, letters, contracts and many other that require proof of authorship. By signing, a person provides potentially unique information regarding the way she or he converts intentions into spontaneous hand movement. Writing speed, traversed path, pen tilt, pressure applied, all these data are articulated (in the form of latent variables) to result in a static figure on signed documents (Nel et al., 2005).

Signature analysis can be divided in two categories: offline and online. In the offline mode, either signatures are available through the traditional *wet ink* method (such as in paper documents), or they are available in scanned form, through optical devices, such as scanners and digital cameras. In both cases, all available data corresponds to static signature images. This type of approach is not the most efficient for verifying signatures due to the fact that relevant dynamic information is discarded.

In the online mode, a person uses a digitizing device (e.g. digitizing tablets or touchscreen devices) to directly record signals from the hand movement. This provides much more information than a static image, for the digitizing device typically can record several complementary signals, such as the path travelled by the pen tip, as well its instantaneous speed, applied pressure and pen tilt. This approach is the one that dominates research on signature verification now, due

to the worldwide spreading of affordable acquisition devices (Plamondon and Srihari, 2000).

However, the offline approach still has some attractive aspects. For instance, even today, many contracts and credit card authorization are performed offline, i.e., through traditional signatures on paper. Indeed, although online signature verification has higher reliability, in many practical situations, for economical or practical reasons, *wet ink* signatures is a useful document authentication method yet. And even in the unlikely scenario of a complete substitution of *wet ink* signatures by electronically acquired ones, at least the task of ancient ink on paper signature verification should remain a relevant topic, due to the large amount of old signed documents, whose authenticity are potentially waiting to be verified (Qiao et al., 2006).

To give some fundamental definitions and jargon, we assume that a signature verification is a process that determines whether a tested signature was produced by a target individual, from which at least one genuine signature is available. If under some chosen similarity criteria the tested signature is similar to the genuine references, below a pre-established *similarity threshold*, it is labeled a *true* signature. Otherwise, the signature is labeled *false*, a *forgery*. Moreover, in (Coetzer et al., 2004), forgeries are classified as: *random forgery*, when a forger does not know the author's name neither the original signature; *simple forgery*, when a forger knows the author's name, but not the original signature; and *skilled forgery*, when both author's name and signature are known.

In general, the offline signature verification pro-

cess can be divided into four steps (Al-Omari et al., 2011): *Acquisition*, *Preprocessing*, *Feature Extraction* and *Comparison*. In the *Preprocessing* step, image quality is improved and pixels are transformed to reduce the computational burden of the subsequent steps. Examples of techniques applied in this step are: thinning, color conversion, noise reduction, smoothing, morphological operations and resizing. For instance, in (Shah et al., 2016), images are cropped to exclude redundant white regions. The *Feature Extraction* step is where most works propose innovations. According to (Batista et al., 2008), an ideal feature extraction technique extracts a minimal feature set that maximizes interpersonal variability amongst signature samples from various subjects, whereas it minimizes intrapersonal variability amongst samples belonging to the same subject. Lee and Pan (Lee and Pan, 1992) divide the features into three classes: *Global Features*, *Local Features* and *Geometrical Features*.

Typical features extracted from offline signatures are marginal projections. Shanker et Rajagopalan (Shanker and Rajagopalan, 2007) extracts vertical projection of bitmaps corresponding to signatures, thus yielding profiles which are compared through Dynamic Time Warping (DTW). Likewise, Coetzer et al. (Coetzer et al., 2004) pushes a bit further the same idea, by using many such marginal projections, over different angles, what they call *Discrete Radon Transform*, whose behaviour is modelled with a Hidden Markov Model. Nguyen et al. (Nguyen et al., 2009) also use similar projections. Indeed, Nguyen et al. use two techniques for global features extraction: the first is derived from the total *energy* a writer uses to create a signature, whereas the second technique employs information from the vertical and horizontal projections of a signature, focusing on the proportion of the distance between key strokes in the image, and the height/width of the signature.

Although marginal projections are more commonly used in literature (possibly because it is less prone to image segmentation mistakes), straightforward approaches to feature extraction may also rely upon image skeletonization (Gonzalez and Woods, 2007). Typically, skeletonization is used to filter foreground pixels in bitmaps. But it can also be used to map offline signatures into sets of points, similar to online representations. This representation is appealing because online verification techniques may be deployed, such as the use of DTW to compare segments of points from different signatures. Indeed, it corresponds to the baseline method implemented in this paper, as explained in Section 3.

Once features are available, signature authenticity

verification can be performed. To simulate actual verification in banks, for instance, most academic works randomly select a small number of genuine signature samples from each user (typically from 5 to 15) to play the role of a set of enrolled signatures. Then, samples from the remaining dataset of false and genuine signatures are randomly taken to simulate verification attempts. These test samples are compared to the enrolled samples, and a decision is made. If a genuine signature is rejected as false, it is called a *false rejection* error. By contrast, if a forgery is accepted as genuine, it is called a *false accept* error. The experimental protocol for False Accept Rate (FAR) and False Reject Rate (FRR) computation used in this work is explained in Section 5.

Furthermore, in this paper, we propose a method inspired by online approaches, through the compact codification of segments of skeletonized online signatures, as explained in Section 4. These skeletonized segments are the basic aspect of this work, and they are separately explained in Section 2. Moreover, for the chosen segmentation induces a straightforward method similar to classical online verification strategies, through DTW, we implement this method to be used as a baseline, as explained in Section 3.

Concerning our experiments, we take into account that in biometrics, comparing performances measures using different databases can be misleading, and is a frequent issue. To allow a direct comparison with a straightforward baseline approaches, under the same database and experimental protocol, we only compare results obtained with the well-known MCYT-100 dataset of online signatures, which are converted into offline (bitmaps) representation, as explained in Section 5, where comparative results are provided, along with our experimental setup. It is worth noting that human baseline performance is presented for the same dataset, in the spirit of works such as (Coetzer et al., 2006) and (Morocho et al., 2016). Finally, in Section 6, we conclude by discussing the usefulness of comparing automatic verification performances do mean opinion performance through Mean Opinion Scores (MOS). This discussion gives a framework to evaluate the usefulness of our method.

2 Unambiguously Connected Skeleton Segments

Raw online signature signals are frequently represented by two vectors of samples: a sequence of regularly sampled horizontal positions, $x_{\text{ONLINE}}(n)$, and another sequence of vertical positions, $y_{\text{ONLINE}}(n)$, through time, where n stands for the sample counter.

As compared to offline representation, signature verification through signals $x_{ONLINE}(n)$ and $y_{ONLINE}(n)$ is significantly better.

Although we know that velocity information may not be completely recovered from offline representations, we address the offline signature verification problem by first recovering horizontal and vertical signals, which may be regarded as pseudo-versions of $x_{ONLINE}(n)$ and $y_{ONLINE}(n)$. This is done through standard skeletonization, as described in (Gonzalez and Woods, 2007). Unlike true online representations, skeletons from each signature, are sets of points whose ordination may be ambiguous. For instance, in Figure 1 a given signature skeleton can be regarded as a connected graph $G(V, E)$ where the vertices V are the points of the skeleton and the edges E are bidirectional connection between neighbouring points (8-connected neighbourhood). Let us also consider that the degree of a vertex V is the number of neighbouring vertices V is connected to.

A straightforward conclusion is that the comparison between two skeletons (two offline signatures) can be preformed through the comparison between sequences of points, as in online comparison. However, it is clear that a lot of ambiguities concerning the sequencing of points turn this problem into a combinatorial optimization problem, whose computational cost may be prohibitive. To significantly reduce this cost, both computational methods presented in this paper decompose offline signatures skeletons into *Unambiguously Connected Skeleton Segments* (UCSS), i.e., as illustrated in Figure 1, each UCSS is a segment, or a sequences of directly connected vertices found between:

- two vertices with degrees greater than 2, or
- an one-degree vertex and a vertex with degree greater than 2, or
- two one-degree vertices.

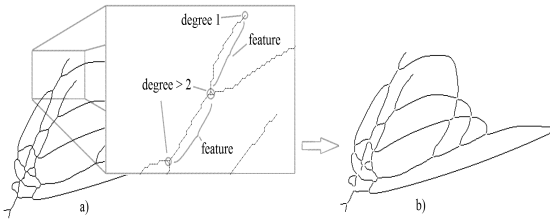


Figure 1: (a) Inside the circles one finds the points of the skeleton that are excluded, and the remaining features are shown in the brackets. (b) Full skeleton with extracted segments.

We assume that each segment, $\{s_{m,n,k}\}$, is a portion of signature where points are unambiguously or-

dered, apart from a single ambiguity in the overall direction of the pen movement. Then, the n -th signature sample, $n = 1, 2, \dots, N$, from the m -th signer, $m = 1, 2, \dots, M$ is represented by a set $\{s_{m,n}\} = \{\{s_{m,n,1}\}, \{s_{m,n,2}\}, \dots, \{s_{m,n,K}\}\}$, where $\{s_{m,n,k}\}$ is the k -th UCSS, or a sequence of $L_{m,n,k}$ pairs of coordinate points, (x_i, y_i) , $1 < i \leq L_{m,n,k}$.

To take into account the single ambiguity in the overall direction of the pen movement, each UCSS is represented twice: first with the sequence of pairs in a given order, (x_i, y_i) , and then the same sequence in reversed order.

3 Baseline Method

Two methods for automatic offline signature verification are used in this paper. The first method is considered as a baseline, for it is a straightforward application of Dynamic Time Warping to compute distances between UCSS. In this method, the standard DTW method under Itakura's restrictions (Itakura, 1975) is applied to systematically compare every segment (and its reverted dual) to every segments of a given reference signature. Consider, for instance, a test signature with K_{test} UCSS, and a single reference signature with K_{ref} UCSS. Then, each UCSS in the reference signature is compared to all K_{test} UCSS from the test signature, and the minimum distance is taken. In other words, each UCSS in a test signature is associated to a single UCSS in the reference signature — the UCSS yielding the minimum DTW distance. More precisely:

$$C_{m,ref}(k) = \min_i \left(\frac{DTW(\{s_{m,test,k}\}, \{s_{m,ref,i}\})}{L_k} \right), \quad (1)$$

where $k \in 1, 2, \dots, K_{test}$, $i \in 1, 2, \dots, K_{ref}$, and ref and $test$ point toward reference and test signatures, respectively. Moreover, $L_k = L_{m,test,k}$ is the length (number of points) of the k -th UCSS and $DTW(\cdot)$ stands for usual Dynamic Time Warping algorithm under Itakura's restriction. Thus, the total distance between a reference signature $\{s_{m,ref}\}$, and a tested signature $\{s_{m,test}\}$ is the average cost

$$\bar{C}_{m,ref}(test) = (1/K_{ref}) \sum_{k=1}^{K_{ref}} C_{m,ref}(k).$$

To compute a score for a test signature, $\{s_{m,test}\}$, we should further consider that, in this work, we randomly take 5 genuine signatures from each individual as reference set, to be denoted as $\{s_{m,ref_1}\}, \{s_{m,ref_2}\}, \dots, \{s_{m,ref_5}\}$. To compare a tested signature to all references at once, we merge all

UCSS from the 5 references into a single larger set, $\{s_{m,ref}\} = \{\{s_{m,ref_1}\}, \{s_{m,ref_2}\}, \dots, \{s_{m,ref_5}\}\}$, with $K_{ref} = K_{ref_1} + K_{ref_2} + \dots + K_{ref_5}$ UCSS. A single cost associated to each tested signature is given by:

$$J_{m,ref}(test) = \frac{\bar{C}_{m,ref}(test)}{(1/5) \sum_{i=1}^5 \bar{C}_{m,ref'}(ref_i)}, \quad (2)$$

where $\bar{C}_{m,ref'}(ref_i)$ stands for a partial normalization score. This score normalization term is obtained by comparing reference $\{s_{m,ref_i}\}$ to the partial reference set ref' , formed by the 4 remaining reference signatures. In other words, the score $J_{m,ref}(test)$ corresponds to the cost $\bar{C}_{m,ref}(test)$ normalized by the mean of 5 costs found by cross-comparing reference signatures through the leave-one-out strategy.

An important drawback of this baseline method is the computation of $K_{ref} \times K_{test}$ DTW distances in order to obtain a score/cost for each tested signature, where K_{ref} is roughly 5 times the number of segments (UCSS) in a genuine signature. As a consequence, this method has a high computational burden.

4 Proposed Method

To significantly alleviate the computational load of the baseline method, we encode each UCSS as an eight-dimensional (8D) vector that roughly represents the shape and the position of the UCSS, plus a scalar corresponding to the UCSS length—in terms of number of points—, as illustrated in Figure 2.

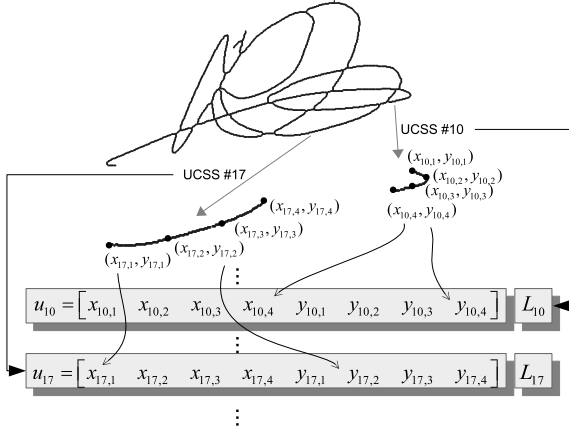


Figure 2: Proposed method: illustration of 8D vectors and lengths of two segments (segments 10 and 17). Each segment (UCSS) is coded with $8 + 1 = 9$ scalars. The last scalar represents the segment length.

This UCSS encoding strategy is the main aspect of the proposed method. We assume that almost all

UCSS are short enough to prevent strong warping, therefore, one may get rid of the high DTW computation cost by replacing each UCSS with 4 sub-sampled points. In other words, it is assumed that UCSS comparisons through DTW are almost equivalent to the much faster Euclidean distance computation between the corresponding 8D vectors (UCSS length is not taken into account when plain DTW is used). Indeed, a UCSS $\{s_{m,n,k}\}$ can be regarded as a composition of two sampled signals, say $x(n)$ and $y(n)$, with $n = 1, 2, \dots, L_{UCSS}$ for a given UCSS, and the proposed coding scheme just takes 4 equally spaced subsamples of $x(n)$ and $y(n)$, thus yielding an 8-dimensional vector $u_{m,n,k}$. For practical purposes, we then assume that:

$$\frac{DTW(\{s_{m,n,i}\}, \{s_{m,n,j}\})}{L_i} \approx \frac{D_2(u_{m,n,i}, u_{m,n,j})}{4},$$

where $D_2(u_{m,n,i}, u_{m,n,j}) = \sum_{k=1}^8 |u_{m,n,i}(k) - u_{m,n,j}(k)|^2$ is the accumulated squared Euclidean distance between the 4 corresponding points in the two compared UCSS. Therefore, the comparison between two signatures is significantly simplified through the use of the following distance, as compared to that in Equation 1:

$$C_{m,ref,k}(test) = \min_i (D_2(u_{m,test,k}, u_{m,ref,i})), \quad (3)$$

and the final score/cost for a given signature is computed as in Equation 2.

5 Experimental Results

In this work, we use the well-known online signatures database MCYT-100, for which error rates (for online verification task) can be abundantly found in literature (Ferrer et al., 2012)(Shah et al., 2016)(Hafemann et al., 2015). This database is a subcorpus of the MCYT database (Ortega-Garcia et al., 2003), acquired from $M = 100$ different writers. Each writer provided 25 genuine signatures, whereas 5 different volunteers provided 25 skilled forgeries per signature ($N = 50$). All signatures were acquired through a Wacom Intuos A6 USB Tablet at constant sampling rate of 100 Hz.

To obtain the offline signatures we need for our experiments, we first convert each MCYT online sample into an image where only horizontal (x_{ONLINE}) and vertical (y_{ONLINE}) pen tip positions through time are considered, as follows:

- (i) Points in each online signature are interpolated using splines to allow for oversampling of the otherwise sparse representation (due to the relatively low sampling rate of 100 samples per second).

- (ii) The oversampled set of points are numerically rounded to integer values, and
- (iii) they are also dilated until segments are approximately four pixels wide.

These three steps are enough to convert the entire on-line data into offline signatures. Indeed, the resulting images are used in the Mean Opinion Score experiment detailed in Subsection 5.1. As for the other experiments, another sequence of steps are still taken to convert offline signatures into pseudo-online ones, namely:

- (i) The standard skeletonization method described in (Gonzalez and Woods, 2007) is applied to each dilated signature image.
- (ii) Resulting skeletons $((x(n), y(n)))$ from each signature sample are centred and scaled, i.e. mean of $x(n)$ and $y(n)$ are both translated to $(0, 0)$, and their corresponding standard deviation are set to 1.

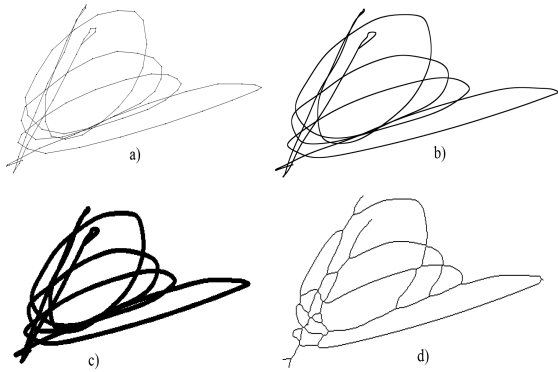


Figure 3: (a) Online signature. (b) Oversampled online signature. (c) Dilated image – approximately four pixels wide lines. (d) Skeleton of the signature.

From the resulting pseudo-online versions of the signatures, represented by a set of coordinate pairs $x(n), y(n)$, UCSS are extracted. It is noteworthy that n no longer stands for a discrete time counter, as in true online representation, but it now represents a mere skeleton point counter, whose correspondence to time ordination is unknown.

In all experiments, to simulate an actual biometric system, we randomly choose 5 genuine signatures from the database to form the enrolment card. Afterwards, the other signatures of the same user are randomly selected and compared with those into the enrolment card. Given a decision threshold, the proportion of true signatures whose costs are above this threshold is the estimated FRR, whereas the proportion of false signatures whose distances are below the threshold is the estimated FAR.

First we compared the computational burden of the two methods to process and compare one enrolment card, with 5 genuine samples per signature, and to compute scores for all $20 + 25 = 45$ remaining signature samples per reference. The processing time for the baseline method was thousands times greater than that for the proposed method. Moreover, the lighter method yielded a significantly better performance, as presented in Figure 4, for a single run of the experiment.

An interesting by-product of both methods is that after two UCSS are associated (according either to Equation 1 or to Equation 3), instead of comparing their shapes, one may just compare their lengths, thus yielding a new score corresponding to the absolute difference of associated lengths. This can be regarded as a third method, here referred to as the *length based* one, in Figure 4 and Table I.

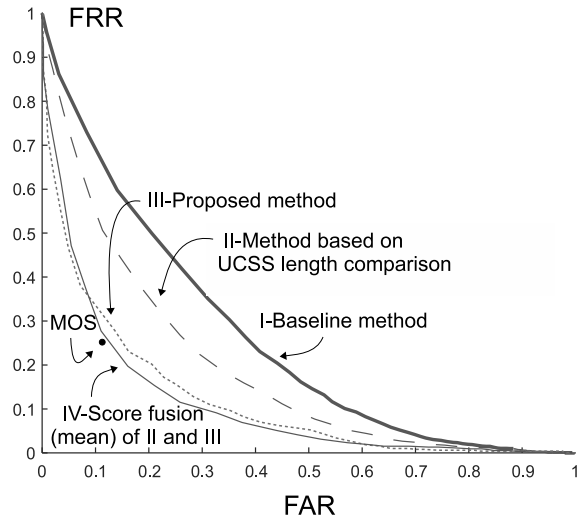


Figure 4: ROC curves for all tested methods. In this set of experiments, we use 5 randomly chosen reference signatures to simulate enrolment, and a pool of 20 remaining genuine signatures, along with 25 false ones to test the simulated system. A black dot also indicates the MOS performance, as explained in Subsection 5.1. Evidently, the decision threshold cannot be known or computationally handled, in this case.

In a second round of experiments, we repeated 50 times, for each method, the following protocol: 5 genuine signatures were randomly chosen as references (enrolment), whereas a set of 10 (5 from the remaining genuine signatures, plus 5 randomly chosen false ones) were used to test the simulated system. For each independent trial, we adjusted the threshold decision to obtain the operational point where FAR equals FRR, the Equal Error Rate (EER). Table I presents all average results per method, in terms of EER and

Table 1: Results for each method after 50 independent runs. Each run corresponds to a random partition of 5 reference and 10 test signatures (5 genuine ones).

Method	EER	std. dev.
I-Baseline	32.2 %	1.4 %
II-Length based	25.1 %	1.2 %
III-Proposed	19.4 %	1.3 %
IV-Fusion (mean) of scores from II and III	18.7 %	1.0 %

its standard deviation over the 50 independent trials.

In both sets of experiments, scores of the two best methods are also fused through simple arithmetic mean, yielding an improved performance.

5.1 Mean Opinion Score Extraction Protocol

To quantify the human performance for the same task, we also prepared cards (one per signature in the MCYT database) with two pages each. These cards contain five genuine signatures on its left side, and ten signatures randomly chosen in its right side. Only 5 out of the 10 signatures on the right side are genuine. An example of these cards can be seen in Figure 5. These cards were presented to different students and lecturers in our university (volunteers), and these volunteers were carefully instructed to study the genuine signatures presented on the left part of the page, and then to label all signatures on the right part by writing in the boxes next to each signature a T, for true (genuine), or a F, for false. No further *a priori* was provided. It is noteworthy that the volunteer did not know how many false signatures there were in the card.

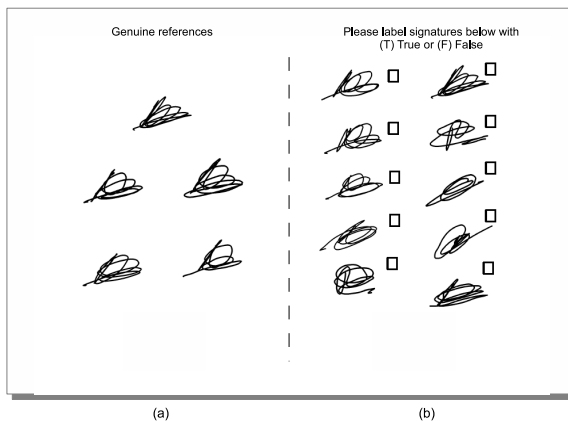


Figure 5: (a) Reference panel with five randomly chosen genuine signatures (b) Panel with five randomly chosen forgeries and five randomly chosen genuine signatures.

A total of 239 cards were filled by 103 volunteers,

and after comparing all provided labels to the true *hidden* labels, the estimated Mean Opinion Score (MOS) obtained was consolidated as:

- 25.6% of False Reject Rate and
- 11.0% of False Accept Rate.

The dot in Figure 4 allows the visual relationship between the rates FAR and FRR from MOS (to which the decision threshold cannot be known or handled) and the ROC curves from the automatic methods, for a range of possible decision thresholds.

6 Conclusion

The main point in this brief work was inspired by the superiority of online signature verification methods. Therefore all methods experimented here are based on skeletonization, possibly the most straightforward method to obtain pseudo-online signature representations from images. Moreover, the baseline method cross-compares all unambiguously connected sequences of points from skeletons, or Unambiguously Connected Skeleton Segments (whose formal definition was given in detail in Section 2). By assuming that UCSS shapes and position are relevant information for biometric verification, we should expect that the systematic cross-comparison of UCSS, through DTW, would yield the best performance, at a high computational cost.

However, it was noticed instead that the alternative method initially proposed to alleviate the high computational burden of the baseline method, by considering only 4 points per UCSS yielded a significantly better performance, as compared to the baseline method. Moreover, even the very simple by-product method based on the comparison between UCSS lengths performed better than the baseline method.

From these rather surprising results, we conjecture that, in average, UCSS shape details are not relevant information carriers for the biometric verification purpose. Indeed, given the comparative performances, we should even conclude that UCSS shape details are disturbing noises, for biometric verification task. This point is an interesting matter for future works.

Regarding the by-product method based on UCSS length, it is noteworthy that the use of either DTW or Euclidean distance to match UCSS is a necessary step. In other words, behind the apparent simplicity of this method, one should be aware that the matching of UCSS is a not so simple step of it.

Interestingly, the superiority of the joint approach where UCSS shape and length are combined may

have a connection to the *lost* signal of pen tip velocity, which in turn is the main signal for biometric verification (Canuto et al., 2016). Indeed, UCSS shape (straightness) and length are expected to be somehow dependent on pen tip velocity, either through two-thirds power law, or through isochrony (Viviani and Flash, 1995), and this dependency is also an attractive subject for future works. For a while, we may just conjecture that the fusion of length and shape based scores is somehow related to inferred velocity signal, given a signature image, which may explain its relatively good performance.

Clearly, even the best performance presented here is far from performances typically found in literature for online verification, with the same database in its original (online) form. By contrast, in this work we assume that, for the offline verification task, the best possible performances are not far from what a crowd of willing and attentive humans can do. Therefore, instead of comparing our results to rates from other works, on similar tasks (but with different datasets), we choose to use a MOS-based rate as basis for comparison, and it indicates that the best performances we obtained are indeed close to that of a crowd of humans. Nonetheless, further experiments using actual offline signature databases and other competing methods are intended to be done in the sequel of this work.

REFERENCES

- Al-Omari, Y. M., Abdullah, S. N. H. S., and Omar, K. (2011). State-of-the-art in offline signature verification system. In *Pattern Analysis and Intelligent Robotics (ICPAIR), 2011 International Conference on*, volume 1, pages 59–64. IEEE.
- Batista, L., Rivard, D., Sabourin, R., Granger, E., and Maupin, P. (2008). State of the art in off-line signature verification. *Pattern Recognition Technologies and Applications: Recent Advances*, pages 39–62.
- Canuto, J., Dorizzi, B., Montalvão, J., and Matos, L. (2016). On the infinite clipping of handwritten signatures. *Pattern Recogn. Lett.*, 79(C):38–43.
- Coetzer, J., Herbst, B., and Du Preez, J. (2006). Off-line signature verification: a comparison between human and machine performance. In *Tenth International Workshop on Frontiers in Handwriting Recognition*. Suvisoft.
- Coetzer, J., Herbst, B. M., and du Preez, J. A. (2004). Offline signature verification using the discrete radon transform and a hidden markov model. *EURASIP Journal on Advances in Signal Processing*, 2004(4):1–13.
- Ferrer, M. A., Vargas, J. F., Morales, A., and Ordóñez, A. (2012). Robustness of offline signature verification based on gray level features. *IEEE Transactions on Information Forensics and Security*, 7(3):966–977.
- Gonzalez, R. C. and Woods, R. E. (2007). Image processing. *Digital image processing*, 2.
- Hafemann, L. G., Sabourin, R., and Oliveira, L. S. (2015). Offline handwritten signature verification-literature review. *arXiv preprint arXiv:1507.07909*.
- Itakura, F. (1975). Minimum prediction residual principle applied to speech recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 23(1):67–72.
- Lee, S. and Pan, J. C. (1992). Offline tracing and representation of signatures. *Systems, Man and Cybernetics, IEEE Transactions on*, 22(4):755–771.
- Morochó, D., Morales, A., Fierrez, J., and Tolosana, R. (2016). Signature recognition: establishing human baseline performance via crowdsourcing. In *2016 4th International Conference on Biometrics and Forensics (IWBF)*, pages 1–6. IEEE.
- Nel, E., du Preez, J. A., and Herbst, B. M. (2005). Estimating the pen trajectories of multi-path static scripts using hidden markov models. In *Document Analysis and Recognition, 2005. Proceedings. Eighth International Conference on*, pages 41–45. IEEE.
- Nguyen, V., Blumenstein, M., and Leedham, G. (2009). Global features for the off-line signature verification problem. In *2009 10th International Conference on Document Analysis and Recognition*, pages 1300–1304. IEEE.
- Ortega-Garcia, J., Fierrez-Aguilar, J., Simon, D., Gonzalez, J., Faundez-Zanuy, M., Espinosa, V., Satue, A., Hernaez, I., Igarza, J.-J., Vivaracho, C., et al. (2003). Mcyt baseline corpus: a bimodal biometric database. *IEE Proceedings-Vision, Image and Signal Processing*, 150(6):395–401.
- Plamondon, R. and Srihari, S. N. (2000). Online and offline handwriting recognition: a comprehensive survey. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(1):63–84.
- Qiao, Y., Nishiara, M., and Yasuhara, M. (2006). A framework toward restoration of writing order from single-stroked handwriting image. *IEEE transactions on pattern analysis and machine intelligence*, 28(11):1724–1737.
- Shah, A. S., Khan, M., Subhan, F., Fayaz, M., and Shah, A. (2016). An offline signature verification technique using pixels intensity levels.
- Shanker, A. P. and Rajagopalan, A. (2007). Off-line signature verification using dtw. *Pattern recognition letters*, 28(12):1407–1414.
- Viviani, P. and Flash, T. (1995). Minimum-jerk, two-thirds power law, and isochrony: converging approaches to movement planning. *Journal of Experimental Psychology: Human Perception and Performance*, 21(1):32.